SiteMinder Federation

파트너 관계 페더레이션 안내서

12.52 SP1

ca technologies

CA Technologies 제품 참조

이 문서는 다음 CA Technologies 제품을 참조합니다 :

- SiteMinder

CA 에 문의

기술 지원팀에 문의

온라인 기술 지원 및 지사 목록, 기본 서비스 시간, 전화 번호에 대해서는 http://www.ca.com/worldwide 에서 기술 지원팀에 문의하십시오.

설명서 변경 사항

SiteMinder 의 이전 릴리스에서 발견된 문제점으로 인해 12.52 설명서의 내용이 업데이트되지 않았습니다.

SiteMinder 의 이전 릴리스에서 발견된 문제점으로 인해 다음과 같은 내용이 12.52 SP1 설명서에서 업데이트되었습니다.

- 세션을 구성하기 위해 인증 URL 보호 (페이지 28) - "IdP 파트너 구성" 아래에 새로운 하위 단원이 추가되었습니다. 이 단계는 사용자에게 세션을 구성하기 위해 인증 URL 을 보호하는 방법을 설명합니다.
- 세션에서 인증 URL 보호 필요 (페이지 65) - 세션을 구성하기 위해 인증 URL 을 보호해야 하는 요구 사항에 대해 설명하는 새 단원이 추가되었습니다. 이 단계는 어설션 당사자 측 구성에 필요합니다.

목차

제 1 장: 파트너 관계 페더레이션 소개

이 섹션은 다음 항목을 포함하고 있습니다.

제품 및 구성 개요

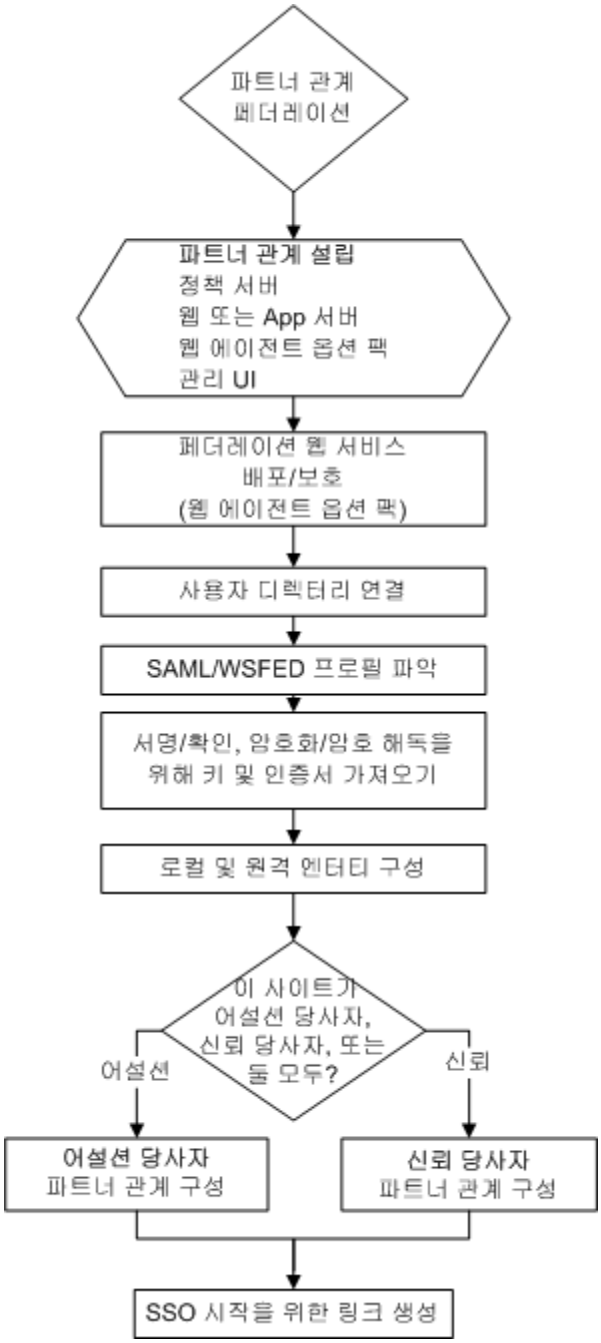
페더레이션된 파트너 관계는 아이덴티티 정보를 유연하고 이식 가능하게 만듭니다. 파트너 관계 페더레이션은 트러스트된 비즈니스 파트너의 네트워크 간에 안전한 싱글 사인온 및 싱글 로그아웃을 제공합니다.

SiteMinder 파트너 관계 페더레이션을 통해 고객은 웹 액세스 관리 시스템과 함께 또는 그와 별개로 페더레이션된 파트너 관계를 유연하게 설정할 수 있습니다. 파트너 관계 페더레이션은 표준 기반 페더레이션을 위한 배포하기 쉬운 솔루션을 제공합니다. 조직은 파트너 관계 페더레이션을 사용하여 어설션 당사자 또는 신뢰 당사자의 역할을 할 수 있습니다. 어설션 당사자는 사용자 인증 및 아이덴티티 어설션을 제공합니다. 신뢰 당사자는 사용자 아이덴티티를 사용하여 웹 리소스와 서비스에 대한 액세스를 허용합니다.

파트너 관계 페더레이션은 다음 프로필을 지원합니다.

- SAML 1.1
- SAML 2.0
- WS-페더레이션

다음 순서도는 파트너 관계 페더레이션을 구성하는 일반적인 프로세스를 보여 줍니다.



프로그래머리스 페더레이션

프로그래머리스 페더레이션은 보안 인증, 사용자 명확성, 조사 및 SAML 어설션의 수정을 가능하게 하는 HTTP 기반 접근법입니다. 프로그래머리스 페더레이션의 장점은 응용 프로그램이 언어별 SDK 또는 다른 바인딩을 사용할 필요 없이 이러한 태스크를 수행할 수 있다는 점입니다.

프로그래머리스 페더레이션은 HTTP/HTTPS 요청 및 응답을 사용합니다. 이러한 요청 및 응답에는 REST(Representational State Transfer) 시스템 아키텍처를 구현한 웹 서비스를 사용하는 URL 및 HTML 기반 프로토콜을 통해 액세스할 수 있습니다.

응용 프로그램은 HTTP 요청을 보내고, HTTP 응답을 읽고, XML 을 구문 분석하여 프로그래머리스 기능의 장점을 활용할 수 있습니다.

프로그래머리스 페더레이션의 핵심 부분은 데이터 교환에 보안을 적용하는 기능입니다. SiteMinder 는 데이터에 보안을 적용하기 위해 개방 형식 쿠키를 사용합니다. 개방 형식 쿠키는 강력한 암호화 알고리즘을 지원하는 잘 정의된 쿠키 형식입니다. 암호화된 쿠키는 SiteMinder 와 로컬 또는 원격 응용 프로그램 간의 응답에 보안을 적용합니다. 이 쿠키는 Perl 또는 Ruby 와 같은 개방 형식 쿠키가 지원하는 것과 동일한 암호화 및 암호 해독 알고리즘을 지원하는 모든 언어로 작성될 수 있습니다.

다음과 같은 파트너 관계 페더레이션 기능이 프로그래머리스 페더레이션을 구현합니다.

위임된 인증

위임된 인증을 통해 SiteMinder 는 타사 WAM(웹 액세스 관리) 시스템을 사용하여 보호된 페더레이션 리소스를 요청하는 모든 사용자의 인증을 수행할 수 있습니다. 타사 WAM 은 인증을 수행한 다음 페더레이션된 사용자 아이덴티티를 SiteMinder 에 전송합니다.

HTTP/HTTPS 요청 및 응답은 프로비저닝을 위한 통신을 용이하게 해줍니다.

신뢰 당사자의 프로비저닝

프로비저닝은 데이터 및 응용 프로그램에 액세스하기 위해 필요한 계정 권한 및 액세스 권한을 가진 클라이언트 계정을 생성하는 프로세스입니다. 파트너 관계 페더레이션 프로비저닝은 사용자를 위한 새 계정을 설정하거나 SAML 어설션에서 전송된 정보를 기존 사용자 계정에 입력할 수 있습니다.

원격 프로비저닝은 SiteMinder 프로비저닝 방법 중 하나입니다. 원격 프로비저닝은 독립적 프로비저닝 응용 프로그램을 사용하여 사용자 레코드를 설정합니다. SiteMinder 는 어설션 데이터를 전달하기 위해 데이터가 포함된 암호화된 쿠키를 생성합니다. 이 쿠키는 사용자 계정 생성을 담당하는 원격 프로비저닝 응용 프로그램으로 전송됩니다.

HTTP/HTTPS 요청 및 응답은 프로비저닝을 위한 통신을 용이하게 해줍니다.

대상 사용자

이 안내서에서는 독자가 다음 개념을 이해하고 있는 것으로 가정합니다.

- 기본 SAML 및 WS-페더레이션 기초 사항
- 페더레이션 바인딩
- SSO(싱글 사인온), SLO(싱글 로그아웃) 및 싱글 사인아웃 등의 페더레이션된 프로필
- PKI(공개 키 인프라) 기초 사항
- SSL(Secure Socket Layer) 통신 기본

이 안내서에서 사용하는 용어

이 가이드에서는 페더레이션된 SAML 과 WS-페더레이션에 관련된 표준 바인딩 및 프로필 용어뿐 아니라 다음 용어도 사용됩니다.

파트너 엔터티 용어

본 안내서에서는 *어설션 당사자*와 *신뢰 당사자*라는 용어를 사용하여 페더레이션된 관계의 양쪽 당사자를 식별합니다.

어설션을 생성하는 당사자를 어설션 당사자라고 합니다. 어설션 당사자는 다음과 같을 수 있습니다.

- SAML 1.x 생산자
- SAML 2.0 IdP(아이덴티티 공급자)
- WS-페더레이션 IP(아이덴티티 공급자)

인증 목적으로 어설션을 소비하는 당사자를 신뢰 당사자라고 합니다. 신뢰 당사자는 다음과 같을 수 있습니다.

- SAML 1.x 소비자
- SAML 2.0 SP(서비스 공급자)
- WS-페더레이션 RP(리소스 파트너)

사이트는 어설션 당사자(생산자/IdP/IP)와 신뢰 당사자(소비자/SP/RP)로 작동할 수 있습니다.

개방 형식 쿠키

사용자 아이덴티티 정보가 포함된 쿠키입니다. 개방 형식 쿠키는 쿠키를 생성하는 방식에 따라 FIPS 또는 비 FIPS 호환 알고리즘을 사용하여 암호화할 수 있습니다. CA SiteMinder® Federation SDK 를 사용하여 개방 형식 쿠키를 생성하거나 UTF-8 인코딩을 지원하는 프로그래밍 언어를 사용하여 수동으로 생성할 수 있습니다.

FIPS 암호화된 개방 형식 쿠키가 필요한 경우 쿠키를 생성하고 읽으려면 SDK 를 사용하십시오. CA SiteMinder® Federation Java SDK 는 FIPS 호환(AES) 알고리즘 또는 비 FIPS(PBE) 알고리즘을 사용하여 쿠키를 암호화할 수 있습니다. CA SiteMinder® Federation .NET SDK 는 FIPS 호환 알고리즘만 사용하여 쿠키를 암호화할 수 있습니다.

UEL(Unified Expression Language)

UEL(Unified Expression Language)은 주로 Java 웹 응용 프로그램을 위한 특수한 Java 식 구문입니다. 웹 페이지에 식을 포함하는 용도로 UEL 을 사용할 수 있습니다. 파트너 관계 페더레이션의 경우 UEL 은 어설션 특성과 신뢰 당사자의 응용 프로그램 특성 간에 매핑을 정의하는 데 사용해야 하는 언어입니다.

파트너 관계 페더레이션 대화 상자 탐색

관리 UI 에서는 파트너 관계 페더레이션 개체를 생성하고 수정하기 위한 구성 마법사를 제공합니다. 구성 마법사의 단계를 따라 개체의 구성 단계를 탐색하십시오.

제 2 장: 파트너 관계 페더레이션에 대한 사전 요구 사항

이 섹션은 다음 항목을 포함하고 있습니다.

SiteMinder 어설션 파트너에 대한 사전 요구 사항

SiteMinder 를 어설션 파트너로 사용하려면 다음 조건을 확인하십시오.

- 정책 서버가 설치되어 있어야 합니다.
- 웹 에이전트 및 웹 에이전트 옵션 팩이 설치되어 있어야 합니다. 웹 에이전트는 사용자를 인증하고 SiteMinder 세션을 설정합니다. 옵션 팩은 페더레이션 웹 서비스 응용 프로그램을 제공합니다. 적절한 네트워크 시스템에 FWS 응용 프로그램을 배포해야 합니다.

  자세한 내용은 *웹 에이전트 옵션 팩 안내서*를 참조하십시오.
- 메시지 서명 및 암호 해독이 필요한 기능을 위해 개인 키와 인증서를 가져와야 합니다.
- 파트너 관계에 대한 사용자 디렉터리로 ODBC 데이터베이스를 선택하기 전에 SQL 쿼리 체계와 올바른 SQL 쿼리를 설정해야 합니다. 이 사전 요구 사항은 ODBC 를 사용하려는 경우에만 필요합니다.
- 페더레이션된 네트워크 내에 신뢰 파트너가 설정되어 있어야 합니다.

SiteMinder 신뢰 파트너에 대한 사전 요구 사항

SiteMinder 가 신뢰 파트너의 역할을 하려면 다음 요구 사항이 충족되어야 합니다.

- 정책 서버가 설치되어 있어야 합니다.
- 웹 에이전트 및 웹 에이전트 옵션 팩. 웹 에이전트는 사용자를 인증하고 SiteMinder 세션을 설정합니다. 옵션 팩은 페더레이션 웹 서비스 응용 프로그램을 제공합니다. 적절한 네트워크 시스템에 FWS 응용 프로그램을 배포해야 합니다.

  자세한 내용은 *웹 에이전트 옵션 팩 안내서*를 참조하십시오.
- 메시지 서명 및 암호화를 필요로 하는 기능을 위해 개인 키와 인증서를 가져옵니다.
- 어설션 파트너는 페더레이션된 네트워크 내에서 설정됩니다.

제 3 장: 간단한 파트너 관계를 사용하여 시작

이 섹션은 다음 항목을 포함하고 있습니다.

기본 SAML 2.0 파트너 관계

파트너 관계 페더레이션을 시작하는 한 가지 방법은 파트너 관계를 구성하는 것입니다. 이 장에서는 기본 SAML 2.0 페더레이션 파트너 관계인 SAML 2.0 POST 프로필을 사용하여 싱글 사인온을 설정하는 방법을 설명합니다. 기본 구성으로 시작하여 최소 개수의 단계를 완료하면 파트너 관계 페더레이션의 작동 방식을 확인할 수 있습니다.

참고: 이 파트너 관계는 SAML 2.0 에 중점을 두지만 전체 프로세스는 SAML 1.1 과 동일합니다. 파트너 관계의 각 단계에서 구성 설정은 SAML 프로토콜에 따라 다를 수 있습니다.

또한 이 장에서는 실제 프로덕션 환경을 반영하여 디지털 서명 및 싱글 로그아웃과 같은 추가 기능의 구성도 설명합니다. 구성에 아티팩트 바인딩을 추가할 수도 있습니다.

이 장에서 사용하는 샘플 네트워크에는 파트너 관계인 두 사이트 모두에 SiteMinder 가 설치되어 있는 것으로 가정합니다. 하지만 한 사이트에는 SiteMinder 를 설치하고 다른 사이트에는 다른 SAML 호환 제품을 설치하여 파트너 관계를 설정할 수도 있습니다.

두 사이트 모두에 SiteMinder 가 있으면 파트너 관계를 구성하는 관점을 이해해야 합니다. 완전한 파트너 관계를 구성하려면 각 사이트의 통신 방향마다 하나씩, 각 사이트에서 *파트너 관계 정의*를 정의하는 것부터 시작하십시오. 예를 들어 로컬 사이트가 아이덴티티 공급자(IdP)인 경우 로컬 IdP-원격 SP 파트너 관계를 구성하십시오. 이 구성은 하나의 파트너 관계 정의입니다. 파트너 관계 구성을 완료하려면 로컬 SP 에서 역방향의 로컬 SP-원격 IdP 파트너 관계를 구성하십시오.

파트너 관계 정의는 항상 로컬과 원격 엔터티를 구분합니다. 로컬 엔터티는 파트너 관계 페더레이션를 구성하는 사이트의 엔터티입니다. 이 환경은 꼭 SiteMinder 가 설치된 것과 동일한 환경일 필요는 없지만 동일한 도메인이어야 합니다. 원격 엔터티는 파트너 관계 페더레이션를 구성하는 도메인과 다른 도메인에 있는 파트너의 엔터티입니다.

다음 프로세스는 SiteMinder 가 두 사이트 모두에 있을 때 기본 파트너 관계를 생성하는 단계를 보여 줍니다.

1. 사용자 디렉터리 연결을 설정합니다.
2. 세션을 구성하기 위해 인증 URL 을 보호합니다.
3. 로컬 및 원격 엔터티를 생성합니다.
4. IdP 에서 로컬 IdP-SP 파트너 관계 정의를 구성합니다.
5. SP 에서 로컬 SP-IdP 파트너 관계 정의를 구성합니다.
6. 파트너 관계를 활성화합니다.
7. 파트너 관계를 테스트합니다.

샘플 페더레이션 네트워크

처음 생성하는 파트너 관계는 다음의 샘플 네트워크와 같습니다. 절차 및 샘플 네트워크의 URL 은 예제이며 실제 사이트를 나타내지 않습니다.

비즈니스 파트너

- 이름이 IdP1 인 아이덴티티 공급자
- 이름이 SP1 인 서비스 공급자

SAML 프로필 및 기능

- POST 프로필을 사용하는 SAML 2.0
- 싱글 사인온
- 서명 처리 없음
- FIPS_COMPAT 모드

IdP 의 SSO 서비스 URL

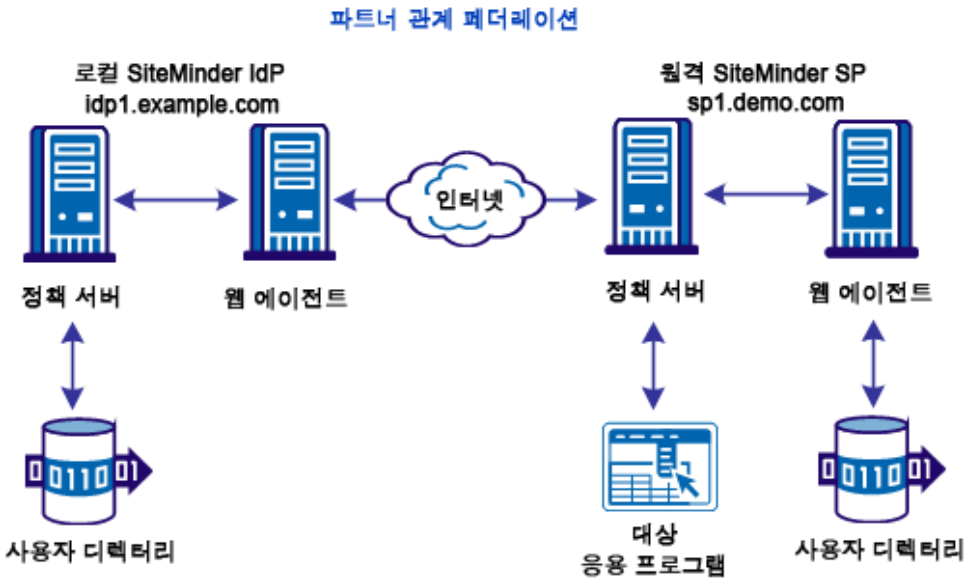
http://idp1.example.com:9090/affwebservices/public/saml2sso

SP 의 어설션 소비자 서비스 URL

http://sp1.demo.com:9091/affwebservices/public/saml2assertionconsumer

참고: 이 샘플 네트워크를 구성하려면 SiteMinder 가 설치된 두 개의 시스템이 필요합니다.

다음 그림은 두 파트너 모두에 SiteMinder 가 있는 샘플 파트너를 보여 줍니다.



필수 구성 요소가 설치되었는지 확인

파트너 관계 페더레이션을 사용하려면 다음 구성 요소가 필요합니다.

- 정책 서버
- 관리 UI
- 웹 에이전트
- 웹 에이전트 옵션 팩

  웹 에이전트 옵션 팩은 FWS(페더레이션 웹 서비스) 응용 프로그램을 포함합니다. 페더레이션에는 FWS 가 필요합니다.

  웹 에이전트 옵션 팩을 설치하고 FWS 를 배포하려면 *웹 에이전트 옵션 팩 안내서*를 참조하십시오.

이 간단한 파트너 관계 배포 예제에서는 이러한 구성 요소가 설치되어 제대로 작동하고 있다고 가정합니다.

IdP 파트너 구성

다음 구성 프로세스는 IdP1 의 관리자 관점에서 작성된 것입니다. 따라서 IdP1 은 로컬 IdP 입니다.

다음은 IdP 파트너를 설정하는 프로세스입니다.

1. 관리 UI 에 로그인합니다.
2. 사용자 디렉터리 연결을 설정합니다.
3. IdP 및 SP 엔터티를 식별합니다.
4. SAML2 IdP->SP 파트너 관계를 생성합니다.
5. 파트너 관계 마법사를 따라 최소한의 필수 설정을 구성합니다.

IdP 에서 사용자 디렉터리 연결 설정

파트너 관계를 설정하려면 먼저 사용자 디렉터리에 대한 연결을 정의해야 합니다. IdP 사용자 디렉터리는 아이덴티티 공급자가 어설션을 생성하는 사용자 레코드로 구성됩니다.

다음 단계에서는 관리 UI 에서 사용자 디렉터리를 구성하는 방법을 지정합니다. 이름이 IdP LDAP 인 디렉터리에 user1 및 user2 가 포함되어 있습니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "인프라", "디렉터리", "사용자 디렉터리"를 차례로 선택합니다.
3. "사용자 디렉터리 만들기"를 클릭합니다.

   "사용자 디렉터리" 대화 상자가 열립니다.
4. 다음 필드를 작성하십시오.

   이름

   IdP LDAP

   네임스페이스

   LDAP

   서버

   www.idp.demo:42088

5. "LDAP 설정" 섹션의 다음 필드에 데이터를 입력합니다.

   루트

   dc=idp,dc=demo

   다른 값의 경우 기본값을 적용합니다.

   "LDAP 사용자 DN 조회"의 다음 필드에 데이터를 입력합니다.

   시작

   uid=

   끝

   ,ou=People,dc=idp,dc=demo

6. "콘텐츠 보기"를 클릭하여 디렉터리 콘텐츠를 볼 수 있는지 확인합니다.
7. "제출"을 클릭합니다.

세션을 구성하기 위해 인증 URL 보호

정책 서버가 어설션을 생성하려면 사용자가 IdP 정책 서버에 세션이 있어야 합니다. 세션을 구성하려면 사용자에게 인증 챌린지가 표시되도록 정책을 사용하여 인증 URL 을 보호하십시오. 그러면 사용자가 로그인하고 세션이 구성됩니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "인프라", "에이전트", "에이전트 만들기"를 차례로 선택합니다.

   "Agent1"이란 이름의 웹 에이전트를 만드십시오.

3. "정책", "도메인", "도메인", "도메인 만들기"를 선택합니다.

   인증 URL 에 대한 정책 도메인을 만듭니다. 챌린지되는 사용자를 포함하는 사용자 디렉터리를 추가하십시오.

4. 정책 도메인에 속한 리소스에 액세스할 수 있어야 하는 사용자를 선택합니다.

5. "영역" 탭을 선택하고 다음 값으로 정책 도메인에 대한 영역을 정의합니다.

   에이전트

   Agent1

   리소스 필터

   /affwebservices/redirectjsp

   기본 리소스 보호

   보호됨

   인증 체계

   기본

   영구 세션

   세션 정보를 저장하려면 HTTP-아티팩트 프로필에 대한 영역 대화 상자의 "세션" 섹션에서 "영구" 확인란을 선택하십시오. 세션 정보는 싱글 로그아웃과 같은 기능 및 특성 기관에 필요합니다.

6. 영역 대화 상자의 "규칙" 섹션에서 "규칙 만들기"를 클릭합니다. 필드에 다음 값을 입력합니다.

   리소스

   /*

   별표는 규칙이 영역의 모든 리소스에 적용됨을 의미합니다.

   허용/거부 및 사용/사용 안 함

   액세스 허용

   "사용" 확인란이 선택됩니다.

   작업

   웹 에이전트 작업

   GET, POST, PUT

7. "정책" 탭을 선택하고 다음 구성 요소를 포함하는 정책을 만듭니다.
   - 사용자 디렉터리에서 선택한 사용자 집합
   - redirectjsp 응용 프로그램 및 관련 규칙을 포함하는 영역

이제 정책이 인증 URL 을 보호합니다.

파트너 관계 엔터티 구성

사용자 디렉터리 연결을 설정한 후에는 파트너 관계의 양측을 식별하십시오. 관리 UI 에서는 각 파트너를 엔터티라고 합니다.

다음 절차에서는 로컬 및 원격 엔터티에 제공할 값을 보여 줍니다. 실제 네트워크 구성에서는 각 측에서 로컬 엔터티를 생성하고, 로컬 엔터티를 메타데이터 파일로 내보낸 다음 파일을 교환할 수 있습니다. 그러면 각 측에서 원격 엔터티를 정의할 수 있습니다.

로컬 IdP 를 생성하려면

1. "페더레이션", "파트너 관계 페더레이션", "엔터티"를 선택합니다.
2. "페더레이션 엔터티 목록"에서 "엔터티 만들기"를 클릭합니다.
3. 엔터티 마법사의 첫 번째 단계에서 다음 사항을 선택하고 "다음"을 클릭합니다.

   엔터티 위치

   로컬

   새 엔터티 유형

   SAML2 IDP
4. 마법사의 두 번째 단계에서 다음 필드에 데이터를 입력하고 "다음"을 클릭합니다.

   엔터티 ID

   idp1

   이 값으로 파트너는 엔터티를 식별합니다.

   엔터티 이름

   idp1

   이 값은 데이터베이스에서 내부적으로 엔터티 개체를 식별합니다. 파트너는 이 값을 인식하지 않습니다.

   기준 URL

   http://idp1.example.com:9090

   다른 설정은 그대로 둡니다.

   참고: "엔터티 이름"의 값은 "엔터티 ID"와 동일할 수 있습니다. 하지만 사이트의 다른 엔터티와 동일한 값을 사용하지 않아야 합니다.

5. 마지막 단계에서 설정을 검토하고 "마침"을 클릭합니다.

"엔터티" 창으로 돌아옵니다.

SP 엔터티를 생성하려면

1. "엔터티" 창에서 시작합니다.
2. "페더레이션 엔터티 목록"에서 "엔터티 만들기"를 클릭합니다.

   "엔터티 만들기" 대화 상자가 표시됩니다.
3. 엔터티 마법사의 첫 번째 단계에서 다음 사항을 선택하고 "다음"을 클릭합니다.

   엔터티 위치

   원격

   새 엔터티 유형

   SAML2 SP
4. 마법사의 두 번째 단계에서 필드에 다음과 같이 데이터를 입력하고 "다음"을 클릭합니다.

   엔터티 ID

   sp1

   이 값으로 파트너는 엔터티를 식별합니다.

   엔터티 이름

   sp1

   이 값은 데이터베이스에서 내부적으로 엔터티 개체를 식별합니다. 파트너는 이 값을 인식하지 않습니다.

어설션 소비자 서비스 URL

인덱스

0

바인딩

HTTP-POST

URL

http://sp1.demo.com:9091/affwebservices/public/saml2assertionconsumer

기본값

항목에 대한 확인란을 선택합니다.

다른 설정은 그대로 둡니다.

5. 마지막 단계에서 설정을 검토하고 "마침"을 클릭합니다.

원격 SP 엔터티가 구성됩니다.

로컬 및 원격 엔터티가 구성된 후 파트너 관계를 생성하십시오.

IdP-SP 파트너 관계 생성

페더레이션 엔터티를 생성한 후에는 파트너 관계 마법사에 따라 IdP ->SP 파트너 관계를 구성하십시오. 마법사는 기본 파트너 관계 매개 변수로 시작합니다.

다음 단계를 수행하십시오.

1. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.
2. "파트너 관계 만들기"를 클릭합니다.
3. "SAML2 IdP -> SP"를 선택합니다.

   이 옵션을 선택하는 것은 현재 로컬 IdP 라는 것을 의미합니다.

   파트너 관계 마법사의 첫 번째 단계가 나타납니다.

4. 필드에 다음 값을 입력합니다.

   파트너 관계 이름

   TestPartnership

   로컬 IDP ID

   idp1

   (풀다운 목록에서 선택)

   원격 SP ID

   sp1

   (풀다운 목록에서 선택)

   기준 URL

   http://idp1.example.com:9090

   차이 시간(초)

   기본값을 적용합니다.

5. IDP LDAP 디렉터리를 "사용 가능한 디렉터리" 목록에서 "선택한 디렉터리" 목록으로 이동합니다.
6. "다음"을 클릭하여 "페더레이션 사용자" 단계로 이동합니다.

어설션 생성을 위한 페더레이션 사용자 지정

"페더레이션 사용자" 대화 상자에서 IdP 가 어설션을 생성할 사용자를 선택합니다.

다음 단계를 수행하십시오.

1. 기본값을 적용합니다.
2. "다음"을 눌러 계속합니다.

기본값을 적용하면 SiteMinder 가 사용자 디렉터리의 모든 사용자에 대해 어설션을 생성할 수 있도록 지정하게 됩니다.

어설션에 이름 ID 추가

"어설션 구성" 단계에서는 NameID 의 형식 및 값과 사용자를 식별하는 특성을 지정할 수 있습니다. 이러한 특성은 어설션에 포함됩니다.

참고: NameID 는 항상 어설션에 포함됩니다.

이 구성에서는 이름 ID 만 지정하십시오. 다른 특성을 추가하지 마십시오.

다음 단계를 수행하십시오.

1. "어설션 구성" 단계에서 다음 필드에 대한 값을 입력합니다.

   이름 ID 형식

   지정되지 않음

   이름 ID 유형

   정적

   값

   GeorgeC
2. "다음"을 클릭하여 SSO(싱글 사인온)를 설정합니다.

IdP 에서 싱글 사인온 설정

파트너 간에 싱글 사인온을 설정하려면 SSO 설정을 구성하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 SSO 및 SLO 단계부터 시작합니다.
2. "인증" 섹션에서 다음 항목을 지정합니다.

   인증 모드

   로컬

   인증 URL

   http://webserver1.example.com/affwebservices/redirectjsp/redirect.jsp

   이 예에서 webserver1 은 웹 에이전트 옵션 팩이 있는 웹 서버를 식별합니다. redirect.jsp 파일은 아이덴티티 공급자 사이트에 설치된 웹 에이전트 옵션 팩에 포함되어 있습니다.

   중요! 액세스 제어 정책을 사용하여 인증 URL 을 보호하십시오.

AuthnContext 구성

기본값을 적용합니다.

인증 클래스

기본값을 적용합니다.

3. "SSO" 섹션에서 다음 항목을 지정합니다.

SSO 바인딩

HTTP-POST

어설션 소비자 URL

http://sp1.demo.com:9091/affwebservices/public/saml2assertionconsumer

4. "다음"을 클릭하여 "서명 및 암호화" 단계로 이동합니다.

서명 처리 사용 안 함

이 간단한 파트너 관계에서는 서명 처리가 사용되지 않도록 지정하십시오. 하지만 프로덕션 환경에서는 아이덴티티 공급자가 어설션에 서명해야 합니다.

다음 단계를 수행하십시오.

1. "서명 및 암호화" 단계에서 "서명 처리 사용 안 함"을 선택합니다.
2. "다음"을 클릭하여 다음 단계로 이동합니다.

IdP-SP 파트너 관계 설정 확인

페더레이션 파트너 관계의 한쪽에 대한 파트너 관계 정의를 완료했습니다. 이제 설정을 확인하십시오.

다음 단계를 수행하십시오.

1. "확인" 대화 상자에서 파트너 관계에 대한 설정을 검토합니다.
2. 설정을 수정하려면 원하는 섹션에서 "수정"을 클릭합니다.
3. 원하는 대로 구성되었으면 "마침"을 클릭합니다.

파트너 관계의 IdP 측이 완료되었습니다. IdP 시스템이 아닌 다른 시스템에서 파트너 관계의 SP 측을 정의하십시오.

SP 파트너 구성

다음 구성 프로세스는 SP(이 예에서는 SP1)의 관리자 관점에서 작성된 것입니다. 따라서 SP1 은 로컬 SP 입니다.

다음은 SP 파트너를 설정하는 프로세스입니다.

1. 관리 UI 에 로그인합니다.
2. 사용자 디렉터리 연결을 설정합니다.
3. IdP 및 SP 엔터티를 식별합니다.
4. SAML2 SP->IdP 파트너 관계를 생성합니다.
5. 파트너 관계 마법사를 따라 최소한의 필수 설정을 구성합니다.

SP 에서 사용자 디렉터리 연결 설정

SP 사용자 디렉터리는 서비스 공급자가 인증에 사용하는 사용자 레코드로 구성됩니다. 다음 단계에서는 관리 UI 에서 사용자 디렉터리를 구성하는 방법을 지정합니다. 이름이 SP LDAP 인 디렉터리에 사용자 user1 및 user2 가 포함되어 있습니다.

사용자 디렉터리를 구성하려면

1. 관리 UI 에 로그인합니다.
2. "인프라", "디렉터리", "사용자 디렉터리"를 차례로 선택합니다.
3. "사용자 디렉터리 만들기"를 클릭합니다.

   "사용자 디렉터리" 대화 상자가 열립니다.
4. 다음 필드를 완료합니다.

   이름

   SP LDAP
5. "디렉터리 설정" 섹션의 다음 필드에 데이터를 입력합니다.

   네임스페이스

   LDAP

   서버

   www.sp.demo:32941

6. "LDAP 검색" 섹션의 다음 필드에 데이터를 입력합니다.

   루트

   dc=sp,dc=demo

   다른 값의 경우 기본값을 적용합니다.

7. "LDAP 사용자 DN 조회" 섹션의 다음 필드에 데이터를 입력합니다.

   시작

   uid=

   끝

   ,ou=People,dc=sp,dc=demo

8. "콘텐츠 보기"를 클릭하여 디렉터리 콘텐츠를 볼 수 있는지 확인합니다.
9. "제출"을 클릭합니다.

파트너 관계 엔터티 식별

사용자 디렉터리 연결을 설정한 후에는 파트너 관계의 로컬 및 원격 측을 식별하십시오. 관리 UI 에서는 각 파트너를 엔터티라고 합니다.

다음 절차에서는 로컬 및 원격 엔터티에 제공할 값을 보여 줍니다. 일반적으로 각 측에서 로컬 엔터티를 생성하고, 로컬 엔터티를 메타데이터 파일로 내보낸 다음 파일을 교환합니다. 그러면 각 측에서 원격 엔터티를 정의할 수 있습니다.

로컬 SP 를 생성하려면

1. "페더레이션", "파트너 관계 페더레이션", "엔터티"를 선택합니다.
2. "엔터티 만들기"를 클릭합니다.
3. 엔터티 마법사의 첫 번째 단계에서 다음 사항을 선택하고 "다음"을 클릭합니다.

   엔터티 위치

   로컬

   새 엔터티 유형

   SAML2 SP

4. 두 번째 단계에서 필드에 다음과 같이 데이터를 입력하고 "다음"을 클릭합니다.

   엔터티 ID

   sp1

   이 값으로 파트너는 엔터티를 식별합니다.

   엔터티 이름

   sp1

   이 값은 데이터베이스에서 내부적으로 엔터티 개체를 식별합니다. 파트너는 이 값을 인식하지 않습니다.

   기준 URL

   http://sp1.demo.com:9091

   참고: 엔터티 ID 와 이름은 아이덴티티 공급자에서 원격 SP 엔터티에 대해 지정한 것과 동일해야 합니다.

5. 설정을 검토하고 "마침"을 클릭합니다.

"엔터티" 창으로 돌아옵니다. 이제 원격 파트너를 구성합니다.

원격 IdP 를 생성하려면

1. "엔터티" 창에서 시작합니다.
2. "엔터티 만들기"를 클릭합니다.
3. 엔터티 마법사의 첫 번째 단계에서 다음 사항을 선택하고 "다음"을 클릭합니다.

   엔터티 위치

   원격

   새 엔터티 유형

   SAML2 IDP

4. 마법사의 두 번째 단계에서 필드에 다음과 같이 데이터를 입력합니다.

   엔터티 ID

   idp1

   이 값으로 파트너는 엔터티를 식별합니다.

   엔터티 이름

   idp1

   이 값은 데이터베이스에서 내부적으로 엔터티 개체를 식별합니다. 파트너는 이 값을 인식하지 않습니다.

   참고: 엔터티 ID 와 이름은 아이덴티티 공급자 측과 동일해야 합니다.

   SSO 서비스 URL 그룹 섹션

   바인딩

   HTTP-리디렉션

   URL

   http://idp1.example.com:9090/affwebservices/public/saml2sso

5. 설정을 검토하고 "마침"을 클릭합니다.

로컬 엔터티 및 원격 엔터티가 구성된 후에는 파트너 관계를 생성할 수 있습니다.

SP-IdP 파트너 관계 생성

파트너 관계 엔터티를 생성한 후에는 파트너 관계 마법사에 따라 SP-> IdP 파트너 관계를 구성하십시오.

다음 단계를 수행하십시오.

1. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.
2. "파트너 관계 만들기"를 클릭합니다.
3. "SAML2 SP->IdP"를 선택합니다.

   파트너 관계 마법사의 첫 번째 단계가 나타납니다.

4. 필드에 다음 값을 입력합니다.

   파트너 관계 이름

   DemoPartnership

   로컬 SP ID

   sp1

   원격 IDP ID

   idp1

   기준 URL

   http://sp1.demo.com:9091

   차이 시간(초)

   기본값을 적용합니다.

5. SP LDAP 디렉터리를 "사용 가능한 디렉터리"에서 "선택한 디렉터리"로 이동합니다.
6. "다음"을 클릭하여 "사용자 ID" 단계로 이동합니다.

사용자 ID 특성 지정

어설션에서 사용자를 식별하는 특성을 지정하십시오. SiteMinder 에서는 아이덴티티 특성 값을 사용하여 SP 의 사용자 디렉터리에서 사용자 레코드를 찾습니다.

사용자 ID 특성을 지정하려면

1. "사용자 ID" 단계로 이동합니다.
2. "어설션에서 아이덴티티 특성 선택" 섹션에서 기본값인 "이름 ID 사용"을 적용합니다.
3. "사용자 디렉터리에 대한 맵 아이덴티티 특성" 섹션에서 다음 항목을 지정합니다.

   LDAP 검색 사양

   uid=%s

   이 항목은 SiteMinder 에 변수(%s)를 어설션의 이름 ID 특성 값으로 바꾸도록 지시합니다. 그러면 SiteMinder 는 값을 샘플 사용자 데이터베이스의 "이름" 열과 비교합니다. 일치하는 항목이 발견되면 사용자의 명확성이 확인되어 대상 리소스에 대한 액세스가 허용됩니다.

4. "페더레이션된 사용자" 섹션에서 기본값을 적용합니다. 사용자 디렉터리의 모든 사용자는 페더레이션된 사용자로 간주됩니다.
5. "다음"을 클릭하여 싱글 사인온을 구성합니다.

SP 에서 싱글 사인온 구성

파트너 간에 싱글 사인온을 설정하려면 SSO 설정을 구성하십시오.

다음 단계를 수행하십시오.

1. SSO 및 SLO 단계에서 시작합니다.
2. SSO 프로필에 대해 "HTTP-POST"를 선택합니다.
3. "원격 SSO 서비스 URL" 섹션에서 다음 값을 지정합니다.

   바인딩

   HTTP-리디렉션

   URL

   http://idp1.example.com:9090/affwebservices/public/saml2sso
4. "서명 및 암호화" 단계에 도달할 때까지 "다음"을 클릭합니다.

   "AuthnContext 구성" 단계는 건너뜁니다.

서명 처리 사용 안 함

이 간단한 파트너 관계에서는 서명 처리가 사용되지 않도록 지정하십시오. 하지만 프로덕션 환경에서는 아이덴티티 공급자가 어설션에 서명해야 합니다.

다음 단계를 수행하십시오.

1. "서명 및 암호화" 단계에서 "서명 처리 사용 안 함"을 선택합니다.
2. "다음"을 클릭하여 다음 단계로 이동합니다.

SP 에서 대상 지정

"응용 프로그램 통합" 단계에서는 대상 리소스를 지정하고 SiteMinder 가 사용자를 대상 리소스로 리디렉션하는 방법을 지정합니다.

다음 단계를 수행하십시오.

1. "리디렉션 모드" 필드에서 "데이터 없음"을 선택합니다.
2. "대상" 필드의 SP 에서 대상 리소스를 지정합니다.

   이 샘플 파트너 관계에서 이 대상은 다음과 같습니다.

   http://spapp.demo.com:80/spsample/welcome.html
3. 대화 상자의 나머지 섹션은 무시합니다.
4. "다음"을 클릭하여 "확인" 단계로 이동합니다.

SP 파트너 설정 확인

페더레이션 파트너 관계의 로컬 SP 측에 대한 파트너 관계를 완료했습니다.

다음 단계를 수행하십시오.

1. "확인" 대화 상자에서 SP 파트너에 대한 설정을 검토합니다.
2. 설정을 수정하려면 해당 섹션에서 "수정"을 클릭합니다.
3. 원하는 대로 구성되었으면 "마침"을 클릭합니다.

파트너 관계의 SP 측이 구성되었습니다.

파트너 관계 활성화

파트너 관계의 양측이 각각 정의되었으므로 이제 파트너 관계를 활성화할 수 있습니다.

SiteMinder 는 파트너 관계의 양측 모두에 설치되므로 IdP 와 SP 에서 파트너 관계를 활성화해야 합니다.

파트너 관계를 활성화하려면

1. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.
2. "페더레이션 파트너 관계 목록"에서 활성화할 항목을 찾습니다. "상태" 열의 값이 "정의됨"인지 확인합니다. 상태가 "미완성"인 경우 파트너 관계를 편집합니다. 필요한 모든 설정이 구성되었는지 확인합니다.
3. 활성화할 파트너 관계 항목 옆의 "작업", "활성화"를 선택합니다.

   "활성화 확인" 대화 상자가 표시됩니다.
4. "예"를 클릭합니다.

   파트너 관계가 활성화되고 "상태" 열의 값이 "활성화"가 됩니다.

파트너 관계 테스트(POST 프로필)

파트너 관계를 구성한 후 두 파트너 간에 싱글 사인온을 테스트합니다.

테스트 작업에는 다음이 포함됩니다.

- 싱글 사인온을 시작할 웹 페이지 생성
- 요청된 페더레이션 리소스 역할을 할 대상 웹 페이지 생성
- 단일 사인 온 테스트

기본 파트너 관계를 테스트한 후 샘플 구성을 추가로 변경할 수 있습니다.

싱글 사인온을 시작할 웹 페이지 생성

테스트를 위하여 싱글 사인온을 시작하는 링크가 있는 HTML 페이지를 직접 생성하십시오. IdP 또는 SP 에서 싱글 사인온을 시작할 수 있습니다. 이 예에서는 SP 에서 시작되는 싱글 사인온을 보여 줍니다.

다음 단계를 수행하십시오.

1. SP 사이트에서 샘플 HTML 페이지를 생성합니다. 다음과 같이 SP 에서 AuthnRequest 서비스에 하드 코딩된 링크를 포함합니다.

   Link to Test POST Single Sign-on

   이 링크는 AuthnRequest 서비스에 사용자를 지정된 아이덴티티 공급자로 리디렉션하여 인증 컨텍스트를 검색하도록 지시합니다.

2. 웹 페이지를 testsso.html 이라는 이름으로 저장합니다.
3. testsso.html 을 웹 서버 문서 루트 디렉터리의 /spsample 하위 폴더 아래에 복사합니다.

   이 샘플 네트워크에서는 대상 웹 서버가 http://spapp.demo:80 입니다.

대상 리소스 생성

싱글 사인온을 테스트하기 위한 마지막 단계는 대상 리소스를 생성하는 것입니다.

다음 단계를 수행하십시오.

1. SP 사이트에 샘플 HTML 페이지를 만들고 다음과 같은 메시지를 포함합니다.

   Welcome to SP1

   Single Sign-on is successful

2. 웹 페이지를 welcome.html 이라는 이름으로 저장합니다.
3. welcome.html 을 웹 서버 문서 루트 디렉터리의 /spsample 하위 폴더 아래에 복사합니다.

   이 샘플 네트워크에서는 대상 웹 서버가 http://spapp.demo.com:80 입니다.

POST 싱글 사인온 테스트

샘플 웹 페이지를 설정한 후에는 싱글 사인온을 테스트하고 파트너 관계 구성이 성공했는지 확인하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계의 양측이 모두 관리 UI 에서 활성화되었는지 확인합니다.
2. 브라우저를 엽니다.
3. 싱글 사인온을 트리거하는 링크가 포함된 웹 페이지의 URL 을 입력합니다. 이 예에서는 다음 URL 을 입력합니다.

   http://spapp.demo.com:80/spsample/testsso.html

   URL 을 입력하면 "Link to Test POST Single Sign-on"이라는 링크가 있는 페이지가 표시됩니다.
4. **Link to Test POST Single Sign-on**(테스트 POST 싱글 사인온 링크)을 클릭합니다.

   싱글 사인온이 시작됩니다. 사용자가 서비스 공급자에서 아이덴티티 공급자로 리디렉션됩니다.

아이덴티티 공급자는 세션을 설정한 후에 사용자를 다시 서비스 공급자의 대상 리소스(welcome.html)로 리디렉션합니다. SP 에서 만든 샘플 시작 페이지가 표시됩니다. 표시된 페이지에 싱글 사인온에 성공했다는 메시지가 표시됩니다.

서명 처리가 사용되도록 설정

SAML 2.0 POST 싱글 사인온에는 디지털 서명된 어설션이 필요합니다. SiteMinder 에서는 서명 및 확인 태스크를 위해 개인 키/인증서 쌍을 사용합니다.

모든 트랜잭션 또는 런타임 작업 전에 IdP1 의 관리자는 인증서(공개 키)가 포함된 SP1 로 파일을 보냅니다. 이 키는 개인 키와 연결됩니다. IdP1 은 공개 키를 사용하여 어설션에 서명합니다. SP1 의 관리자는 인증서를 인증서 데이터 저장소에 추가합니다.

싱글 사인온 트랜잭션이 발생하면 IdP1 은 개인 키를 사용하여 어설션에 서명합니다. SP1 은 어설션을 수신하고 자체 인증서 데이터 저장소의 인증서를 사용하여 어설션 서명을 확인합니다.

IdP 에서 서명 처리 구성

HTTP-POST 싱글 사인온의 경우 Idp1 이 어설션에 서명해야 합니다. IdP 는 인증서 데이터 저장소에 저장된 개인 키를 사용하여 어설션에 서명해야 합니다.

참고: 예에서는 키/인증서 쌍을 가져올 수 있는 파일이 있는 것으로 가정합니다. 또는 개인 키/인증서 쌍이 이미 인증서 데이터 저장소에 있습니다.

서명을 구성하려면

1. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.
2. IdP ->SP 파트너 관계인 TestPartnership 에 대한 항목 옆의 "작업", "비활성화"를 선택합니다.

   편집하기 전에 비활성화해야 합니다.
3. TestPartnership 항목 옆의 "작업", "수정"을 클릭합니다.

   파트너 관계 마법사가 열립니다.
4. "서명 및 암호화" 단계를 선택합니다.
5. "서명" 섹션에서 다음 태스크를 완료합니다.
   a. "서명 처리 사용 안 함"의 선택을 취소합니다.
   b. "서명 개인 키 별칭" 필드 옆에서 "가져오기"를 클릭합니다.

      "인증서/개인 키 가져오기" 창이 열립니다.
6. 다음과 같이 가져오기 마법사를 완료합니다.
   a. 개인 키/인증서 쌍을 가져올 파일을 선택합니다.
   b. pkcs#12 파일의 경우 파일을 암호화하는 암호를 제공합니다. 이 암호는 이전에 설정한 것입니다.
   c. 가져올 파일에서 인증서 항목을 선택하고 "별칭"의 값을 cert1 과 같이 입력합니다.
   d. 선택을 확인하고 "마침"을 클릭합니다.

   "페더레이션 파트너 관계" 목록으로 돌아갑니다.
7. 파트너 관계 항목에 대해 "작업", "수정"을 선택합니다.
8. "서명 및 암호화" 단계로 이동합니다. 이제 가져온 키/인증서를 "서명 개인 키 별칭" 드롭다운 목록에서 사용할 수 있습니다.

9. 별칭 cert1 을 선택하고 "다음"을 클릭합니다.
10. "확인" 대화 상자에서 설정을 검토하고 "마침"을 클릭합니다.

    "파트너 관계" 창으로 돌아옵니다.
11. TestPartnership 항목 옆의 "작업", "활성화"를 선택하여 파트너 관계를 다시 활성화합니다.

이제 IdP 에서 서명 처리가 구성되었습니다.

SP 에서 서명 처리 구성

SP1 에서 어설션의 서명을 확인해야 합니다. 트랜잭션 전에 SP1 은 IdP1 로부터 인증서(공개 키)를 수신했습니다. 이 인증서는 어설션 서명에 사용된 개인 키 IdP1 에 대한 인증서입니다. 이 인증서를 SP1 인증서 데이터 저장소로 가져왔습니다.

서명 확인을 구성하려면

1. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.

   "파트너 관계" 창이 열립니다.
2. DemoPartnership 에 대한 항목 옆의 "작업", "비활성화"를 선택합니다.

   편집하기 전에 비활성화해야 합니다.
3. DemoPartnership 항목 옆의 "작업", "수정"을 클릭합니다.

   파트너 관계 마법사가 열립니다.
4. "서명 및 암호화" 단계를 선택합니다.
5. "서명" 섹션에서 다음 태스크를 완료합니다.
   a. "서명 처리 사용 안 함"의 선택을 취소합니다.
   b. "확인 인증서 별칭" 필드 옆의 "가져오기"를 클릭합니다.

      "인증서/개인 키 가져오기" 창이 열립니다.
6. 다음과 같이 가져오기 마법사를 완료합니다.
   a. 인증서를 가져올 파일을 선택합니다.

b. 가져올 파일에서 인증서 항목을 선택하고 "별칭"의 값을 cert1 과 같이 입력합니다.

c. 선택을 확인하고 "마침"을 클릭합니다.

"페더레이션 파트너 관계" 목록으로 돌아갑니다.

7. 파트너 관계 항목에 대해 "작업", "수정"을 선택합니다.
8. "서명 및 암호화" 단계로 이동합니다. 이제 가져온 키/인증서를 "서명 개인 키 별칭" 드롭다운 목록에서 사용할 수 있습니다.
9. 인증서에 대한 별칭 cert1 을 선택하고 "다음"을 클릭합니다.
10. "확인" 대화 상자에서 설정을 검토하고 "마침"을 클릭합니다.

    "파트너 관계" 창으로 돌아옵니다.

11. DemoPartnership 항목 옆의 "작업", "활성화"를 선택하여 파트너 관계를 다시 활성화합니다.

이제 SP 에서 서명 확인이 구성되었습니다.

싱글 로그아웃 추가

SLO(싱글 로그아웃) 프로토콜을 통해 로그아웃을 초기화한 브라우저에 대한 모든 사용자의 세션이 동시에 종료됩니다. 싱글 로그아웃을 구성하면 권한 없는 사용자가 서비스 공급자의 리소스에 액세스할 수 있도록 열려 있는 세션이 남지 않게 됩니다.

중요! SLO 설정을 보려면 정책 서버 관리 콘솔을 이용하여 세션 저장소가 사용되도록 설정하십시오. 관리 콘솔 사용에 대한 자세한 내용은 *정책 서버 관리 안내서*를 참조하십시오.

IdP 에서 싱글 로그아웃 구성

Idp1 에서 싱글 로그아웃을 구성합니다.

다음 단계를 수행하십시오.

1. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.

   "파트너 관계" 창이 열립니다.

2. TestPartnership 항목 옆의 "작업", "비활성화"를 선택합니다.

   파트너 관계를 편집하기 전에 비활성화합니다.

3. TestPartnership 항목 옆의 "작업", "수정"을 클릭합니다.

   파트너 관계 마법사가 열립니다.

4. SSO 및 SLO 단계를 선택합니다.

5. SLO 섹션에서 다음 필드를 구성합니다.

   SLO 바인딩

   HTTP-리디렉션

   SLO 확인 URL

   http://idp1.example.com:9090/idpsample/SLOConfirm.html

   이 링크는 싱글 로그아웃을 시작한 사이트(이 경우 IdP1)의 확인 페이지입니다. 싱글 로그아웃이 성공적으로 완료되면 사용자가 이 페이지로 리디렉션됩니다.

6. "SLO 서비스 URL" 테이블에서 "행 추가"를 클릭하고 다음 필드에 데이터를 입력합니다.

   SLO 위치 URL

   http://sp1.demo.com:9091/affwebservices/public/saml2slo

   이 링크는 싱글 로그아웃 요청이 원격 SP 로 전송되었음을 나타냅니다.

7. "선택" 열에서 구성한 행을 선택합니다.

8. 마법사에서 "확인" 단계를 클릭하고 구성을 검토합니다.

9. "마침"을 클릭합니다.

   "파트너 관계" 창으로 돌아옵니다.

10. TestPartnership 옆의 "작업", "활성화"를 선택하여 파트너 관계를 다시 활성화합니다.

이제 IdP1 에서 싱글 로그아웃이 구성에 추가되었습니다.

SP 에서 싱글 로그아웃 구성

SP1 에서 싱글 로그아웃을 구성합니다.

SP 에서 싱글 로그아웃을 구성하려면

1. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.

   "파트너 관계" 창이 표시됩니다.

2. "Demo Partnership"(데모 파트너 관계)에 대한 항목 옆의 "작업", "비활성화"를 선택합니다.

   파트너 관계를 편집하기 전에 비활성화합니다.

3. DemoPartnership 에 대한 항목 옆의 "작업", "수정"을 클릭합니다.

   "파트너 관계" 마법사의 첫 번째 단계에 해당하는 대화 상자가 열립니다.

4. "SSO 및 SLO" 단계를 클릭합니다.

5. SLO 섹션에서 다음 필드를 구성합니다.

   SLO 바인딩

   HTTP-리디렉션

   SLO 확인 URL

   http://sp1.demo.com:9091/spsample/SLOConfirm.html

   이 URL 은 로그아웃을 시작한 사이트의 싱글 로그아웃 확인 페이지입니다.

6. "SLO 서비스 URL" 테이블에서 "행 추가"를 클릭하고 다음 필드에 데이터를 입력합니다.

   SLO 위치 URL

   http://idp1.example.com:9090/affwebservices/public/saml2slo

   이 URL 은 싱글 로그아웃 요청이 전송된 위치입니다.

7. "선택" 열에서 구성한 행을 선택합니다.
8. 마법사에서 "확인" 단계를 클릭하고 구성을 검토합니다.
9. "마침"을 클릭합니다.

   "파트너 관계" 창으로 돌아옵니다.
10. "페더레이션 파트너 관계" 목록에서 DemoPartnership 항목 옆의 "작업", "활성화"를 선택하여 파트너 관계를 다시 활성화합니다.

이제 싱글 로그아웃이 SP 에서 구성되었습니다.

싱글 로그아웃 테스트

싱글 로그아웃을 구성한 후에는 이를 테스트하십시오. 이 테스트의 경우 싱글 로그아웃이 SP1 에서 시작됩니다.

SP 에서 싱글 로그아웃을 시작하려면 싱글 로그아웃을 시작하고 확인할 두 개의 웹 페이지가 있어야 합니다.

- welcome.html 을 사용하여 브라우저를 IdP1 의 싱글 로그아웃 서비스로 리디렉션하는 링크를 이 페이지에 추가하십시오. 이 링크의 구문은 다음과 같습니다.

  ```
  <a href="http://idp1.example.com:9090/affwebservices/public/saml2slo>Log Me Out</a>
  ```
- 다음과 같은 로그아웃 확인 메시지가 있는 SLOConfirm.html 이라는 이름의 확인 페이지를 생성하십시오.

  ```
  <p>성공적으로 로그아웃했습니다</p>
  ```

두 페이지를 모두 웹 서버 루트 디렉터리의 하위 폴더 /spsample 아래에 복사하십시오.

참고: SLO 를 테스트할 수 있도록 SSO 트랜잭션을 완료하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계의 양측이 모두 관리 UI 에서 활성화되었는지 확인합니다.
2. 이전에 설명한 지침에 따라 싱글 사인온을 구성하고 테스트합니다.

   싱글 사인온에 성공하면 브라우저에 시작 페이지가 표시됩니다.
3. 브라우저를 열어 두고 시작 페이지에서 **Log Me Out** 링크를 클릭합니다.

   성공하면 다음 메시지가 표시되는 확인 페이지로 리디렉션됩니다.

   성공적으로 로그아웃했습니다.

SSO 에 대한 아티팩트 프로필 설정

기본 파트너 관계는 싱글 사인온에 대한 HTTP-POST 바인딩으로 시작합니다. 하지만 파트너 관계에 SAML 2.0 아티팩트 프로필을 사용할 수 있습니다.

HTTP-아티팩트 바인딩을 구성하는 절차는 마법사의 SSO 및 SLO 단계까지는 POST 바인딩의 절차와 같습니다.

IdP 에서 아티팩트 SSO 구성

이 절차는 SSO 에 대한 HTTP-아티팩트 프로필을 구성하는 방법을 보여 줍니다.

다음 단계를 수행하십시오.

1. 관리 UI 에서 "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.

   "파트너 관계" 창이 표시됩니다.
2. TestPartnership 에 대한 항목 옆의 "작업", "비활성화"를 선택합니다.

   편집하기 전에 비활성화해야 합니다.
3. TestPartnership 에 대한 항목 옆의 "작업", "수정"을 클릭합니다.

   파트너 관계 마법사가 열립니다.
4. "SSO 및 SLO" 단계를 클릭합니다.
5. "인증" 섹션의 기존 설정을 유지합니다.

6. "SSO" 섹션에서 다음 항목을 지정합니다.

   SSO 바인딩

   HTTP-아티팩트

   아티팩트 보호 유형

   파트너 관계

   나머지 설정은 그대로 두십시오.

7. "어설션 소비자 서비스 URL" 테이블에 행을 추가하고 다음 설정을 사용합니다.

   바인딩

   HTTP-아티팩트

   URL

   http://sp1.demo.com:9091/affwebservices/public/saml2assertionconsumer

   이 URL 은 POST 프로필에 사용되는 것과 동일합니다.

8. "백 채널" 섹션에서 "수신 구성"에 대해 다음 인증 방법을 선택합니다.

   인증 방법

   인증 없음

9. 대화 상자의 다른 섹션은 건너뜁니다.
10. "확인" 단계로 이동하여 구성을 검토합니다.
11. "마침"을 클릭하여 구성을 완료합니다.

이제 Idp1 에서 아티팩트 바인딩이 구성되었습니다.

SP 에서 아티팩트 SSO 구성

이 절차는 SSO 에 대한 HTTP-아티팩트 프로필을 구성하는 방법을 보여 줍니다.

다음 단계를 수행하십시오.

1. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.

   "파트너 관계" 창이 표시됩니다.

2. "Demo Partnership"(데모 파트너 관계)에 대한 항목 옆의 "작업", "비활성화"를 선택합니다.

   편집하기 전에 비활성화해야 합니다.

3. DemoPartnership 항목 옆의 "작업", "수정"을 클릭합니다.

   파트너 관계 마법사가 열립니다.

4. "SSO 및 SLO" 단계를 클릭합니다.

5. "SSO" 섹션에서 다음 항목을 지정합니다.

   SSO 프로필

   HTTP-아티팩트

   SSO 서비스 URL

   HTTP-POST 싱글 사인온에 대하여 구성한 것과 동일한 URL 을 유지합니다.

6. "원격 SOAP 아티팩트 레졸루션 URL" 테이블에서 "행 추가"를 클릭합니다. 다음 설정을 입력합니다.

   인덱스

   1

   URL

   http://idp1.example.com:9090/affwebservices/public/saml2ars

7. 테이블의 "선택" 열에서 이 항목을 선택합니다.

8. "백 채널" 섹션에서 "송신 구성"에 대해 다음 인증 방법을 선택합니다.

   인증 방법

   인증 없음

9. "응용 프로그램 통합" 단계에 도달할 때까지 "다음"을 클릭합니다.

SP 에서 대상 지정

"응용 프로그램 통합" 단계에서는 대상 리소스를 지정하고 SiteMinder 가 사용자를 대상 리소스로 리디렉션하는 방법을 지정합니다.

다음 단계를 수행하십시오.

1. "리디렉션 모드" 필드에서 "데이터 없음"을 선택합니다.
2. "대상" 필드의 SP 에서 대상 리소스를 지정합니다.

   이 샘플 파트너 관계에서 이 대상은 다음과 같습니다.

   http://spapp.demo.com:80/spsample/welcome.html
3. 대화 상자의 나머지 섹션은 무시합니다.
4. "다음"을 클릭하여 "확인" 단계로 이동합니다.

파트너 관계 테스트(아티팩트 SSO)

파트너 관계의 양쪽이 작동하면 두 파트너 간에 싱글 사인온을 테스트합니다.

IdP1 이 요청을 받으면 아티팩트를 생성합니다. 그런 다음 아티팩트가 SP1 에 전송됩니다.

SP1 이 아티팩트를 받으면 요청을 다시 IdP1 에 리디렉션합니다. IdP 가 어설션을 검색하여 SP1 에 반환합니다.

싱글 사인온을 시작할 웹 페이지 생성(아티팩트)

테스트를 위하여 싱글 사인온을 시작하는 링크가 있는 HTML 페이지를 직접 생성하십시오. IdP 또는 SP 에서 싱글 사인온을 시작할 수 있습니다. 이 예에서는 SP 에서 시작되는 싱글 사인온을 보여 줍니다.

다음 단계를 수행하십시오.

1. SP 사이트에서 샘플 HTML 페이지를 생성하고 다음과 같이 SP 의 AuthnRequest 서비스에 대한 하드 코딩된 링크를 포함합니다.

   Link for ARTIFACT Single Sign-on

   이 링크는 AuthnRequest 서비스에 사용자를 지정된 아이덴티티 공급자로 리디렉션하여 사용자 인증 컨텍스트를 검색하도록 지시합니다.

2. 웹 페이지를 testartifact.html 이라는 이름으로 저장합니다.
3. testartifact.html 을 웹 서버 문서 루트 디렉터리의 /spsample 하위 폴더 아래에 복사합니다.

   이 샘플 네트워크에서는 대상 웹 서버가 http://spapp.demo:80 입니다.

대상 리소스 생성

싱글 사인온을 테스트하기 위한 마지막 단계는 대상 리소스를 생성하는 것입니다.

다음 단계를 수행하십시오.

1. SP 사이트에 샘플 HTML 페이지를 만들고 다음과 같은 메시지를 포함합니다.

   Welcome to SP1

   Single Sign-on is successful

2. 웹 페이지를 welcome.html 이라는 이름으로 저장합니다.
3. welcome.html 을 웹 서버 문서 루트 디렉터리의 /spsample 하위 폴더 아래에 복사합니다.

   이 샘플 네트워크에서는 대상 웹 서버가 http://spapp.demo.com:80 입니다.

아티팩트 싱글 사인온 테스트

샘플 웹 페이지를 설정한 후에는 싱글 사인온을 테스트하고 파트너 관계 구성이 성공적인지 확인하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계의 양측이 활성화되었는지 확인합니다.
2. 브라우저를 엽니다.
3. 다음과 같이 싱글 사인온을 트리거하는 웹 페이지에 대한 URL 을 입력합니다.

   http://spapp.demo.com:80/spsample/testartifact.html

   참고: 대상 웹 서버는 SiteMinder 가 있는 서버와 다른 서버입니다.

   URL 을 입력하면 "Link to Test ARTIFACT Single Sign-on"(테스트 아티팩트 싱글 사인온 링크)이라는 링크가 있는 페이지가 표시됩니다.
4. **Link to Test ARTIFACT Single Sign-on**(테스트 아티팩트 싱글 사인온 링크)을 클릭하면 싱글 사인온이 시작됩니다.

   사용자가 SP 에서 아이덴티티 공급자로 리디렉션됩니다.

아이덴티티 공급자는 세션을 설정한 후에 사용자를 다시 서비스 공급자의 대상 리소스(welcome.html)로 리디렉션합니다. SP 에서 만든 샘플 시작 페이지가 표시됩니다. 표시된 페이지를 통해 싱글 사인온에 성공했음을 확인할 수 있습니다.

간단한 파트너 관계 이상의 구성 절차

샘플 파트너 관계는 파트너 관계 페더레이션를 사용하여 페더레이션된 파트너 관계를 구성하는 과정을 간략하게 보여 줍니다.

안내서의 나머지 장에서는 수행 가능한 모든 태스크에 대한 자세한 절차를 설명합니다. 자세한 구성 지침에 대해서는 이 절차와 관리 UI 의 도움말을 참조하십시오.

추가 정보:

페더레이션 엔터티 구성 (페이지 69)
파트너 관계 생성 및 활성화 (페이지 81)

제 4 장: 페더레이션 기능에 세션 저장소가 필요함

세션 저장소에는 다음 페더레이션 기능에 대한 데이터가 저장됩니다.

- HTTP-아티팩트 싱글 사인온(SAML 1.x 또는 2.x)

  SAML 어설션 및 연결된 아티팩트가 어설션 당사자 측에서 생성됩니다. 아티팩트가 생성된 어설션을 식별합니다. 어설션 당사자는 신뢰 당사자에 아티팩트를 반환합니다. 신뢰 당사자는 아티팩트를 사용하여 어설션 당사자가 세션 저장소에 저장한 어설션을 검색합니다.

  이 프로세스가 작동하려면 영구 세션이 필요합니다.

  참고: SAML POST 프로필은 세션 저장소에 어설션을 저장하지 않습니다.

- HTTP-POST 단일 사용 정책(SAML 2.0 및 WS-페더레이션)

  단일 사용 정책 기능은 어설션이 신뢰 당사자 측에서 두 번째 세션을 설정하는 데 재사용되지 않도록 합니다. 신뢰 당사자는 자체 세션 저장소에 만료 데이터라고 하는 어설션에 대한 시간 기반 데이터를 저장합니다. 만료 데이터는 어설션이 한 번만 사용되었는지 확인합니다.

  신뢰 당사자에 세션 저장소가 필요하지만 영구 세션은 필요하지 않습니다.

- 싱글 로그아웃(SAML 2.0)

  싱글 로그아웃이 사용되도록 설정된 경우 어느 파트너든지 사용자 세션에 대한 정보를 저장할 수 있습니다. 세션 정보는 세션 저장소에 보관됩니다. 싱글 로그아웃 요청이 완료되면 사용자의 세션 정보가 제거되어 세션이 무효화됩니다.

  아이덴티티 공급자와 서비스 공급자에 영구 세션이 필요합니다.

- 사인아웃(WS-페더레이션)

  사인아웃이 사용되도록 설정된 경우 사용자 컨텍스트 정보가 세션 저장소에 저장됩니다. 이 정보를 통해 정책 서버가 사인아웃 요청을 생성할 수 있습니다. 사인아웃 요청이 완료되면 사용자의 세션 정보가 제거되면서 사용자 세션이 무효화됩니다.

  아이덴티티 공급자와 리소스 파트너에 영구 세션이 필요합니다.

- 인증 세션 변수 유지(모든 프로필)

  신뢰 당사자 측에서 페더레이션을 구성할 때 "인증 세션 변수 유지" 옵션을 선택할 수 있습니다. 이 옵션은 정책 서버에 인증 컨텍스트 데이터를 세션 저장소에 세션 변수로 저장하도록 지시합니다. 정책 서버는 이러한 변수에 액세스하여 인증 결정에 사용할 수 있습니다.

- 어설션 특성 유지(모든 프로필)

  신뢰 당사자 측에서 "특성 유지"를 리디렉션 모드로 선택할 수 있습니다. 리디렉션 모드는 사용자를 대상 응용 프로그램으로 리디렉션하는 방법을 결정합니다. 이 모드에서는 정책 서버가 어설션 특성을 세션 저장소에 저장하여 이 특성을 HTTP 헤더 변수로 제공할 수 있도록 합니다.

- 인증 요청 POST 바인딩(SAML 2.0)

  IdP 가 HTTP-POST 바인딩을 사용하여 전달된 인증 요청을 처리할 수 있기 위해서는 IdP 가 세션 저장소에 요청을 저장해야 합니다.

세션 저장소가 이러한 유형의 사용자 세션, 어설션, 만료 데이터를 저장할 수 있도록 합니다.

세션 저장소가 사용되도록 설정

싱글 사인온, 싱글 로그아웃을 위해 SAML 아티팩트를 사용하고 정책의 단일 사용을 가능하게 할 때는 데이터를 저장하는 세션 저장소가 사용되도록 설정하십시오.

정책 서버 관리 콘솔에서 세션 저장소가 사용되도록 설정하십시오.

다음 단계를 수행하십시오.

1. 정책 서버 관리 콘솔에 로그인합니다.
2. "데이터" 탭을 선택합니다.
3. "데이터베이스" 필드의 드롭다운 목록에서 "세션 저장소"를 선택합니다.
4. "저장소" 필드의 드롭다운 목록에서 사용 가능한 저장소 유형을 선택합니다.

5. "세션 저장소 사용" 확인란을 선택합니다.

   하나 이상의 영역에서 영구 세션을 사용하려면 세션 서버가 사용되도록 설정하십시오. 사용되도록 설정된 경우 세션 서버는 정책 서버 성능에 영향을 줍니다.

   참고: "정책 저장소 데이터베이스 사용" 옵션은 사용되지 않도록 설정됩니다. 성능상의 이유로, 세션 서버와 정책 저장소를 동일한 데이터베이스에서 실행할 수 없습니다.

6. 선택한 저장소 유형에 적절한 데이터 원본 정보를 지정합니다.
7. "확인"을 클릭하여 설정을 저장하고 콘솔을 종료합니다.
8. 정책 서버를 중지했다가 다시 시작합니다.

공유 세션 저장소가 필요한 환경

다음 기능을 사용하려면 SAML 어설션과 사용자 세션 정보를 저장할 공유 세션 저장소가 필요합니다.

클러스터된 정책 서버 환경에서 이러한 기능을 구현하려면 환경을 다음과 같이 설정하십시오.

- HTTP-POST 단일 사용 정책을 *제외한* 모든 기능에 대해 영구 세션의 로그인 영역을 구성하십시오.

  영구 세션은 영역 구성의 일부입니다.

- HTTP-아티팩트 싱글 사인온의 경우 클러스터의 모든 정책 서버에서 생산자/아이덴티티 공급자 사이트의 세션 저장소를 공유하십시오.

  세션 저장소를 공유하면 모든 정책 서버가 각각 어설션에 대한 요청을 받을 때 어설션에 액세스할 수 있게 됩니다.

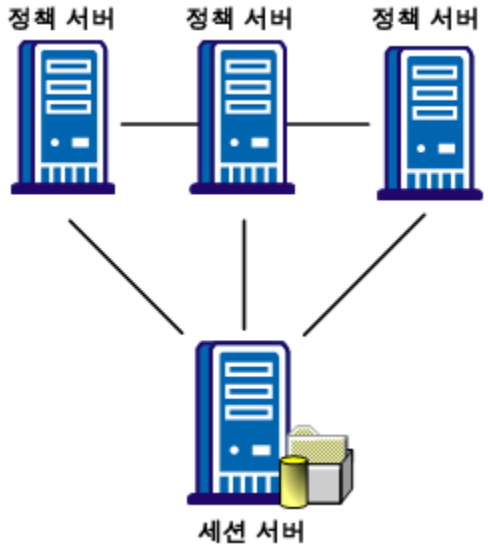
- SAML 2.0 싱글 로그아웃 및 WS-페더레이션 사인아웃의 경우 클러스터의 모든 정책 서버에서 어설션 당사자 및 신뢰 당사자의 세션 저장소를 공유합니다.

  세션 저장소를 공유하면 모든 정책 서버가 각각 세션 로그아웃에 대한 요청을 받을 때 사용자 세션 데이터에 액세스할 수 있게 됩니다.

- HTTP-POST 및 WS-페더레이션 단일 사용 정책 기능의 경우 클러스터에 있는 모든 정책 서버에서 신뢰 당사자의 세션 저장소를 공유합니다.

어설션을 생성 또는 소비하거나 영구 SMSESSION 쿠키를 처리하는 모든 정책 서버는 공용 세션 저장소에 연결할 수 있어야 합니다. 예를 들어 사용자가 example.com 에 로그인하고 해당 도메인에 대한 영구 세션 쿠키를 얻는다고 가정합니다. 이 경우 example.com 에 대한 요청을 처리하는 모든 정책 서버는 세션이 여전히 유효한지 확인할 수 있어야 합니다.

다음 그림에서는 세션 저장소 하나와 통신하는 정책 서버 클러스터를 보여 줍니다.



세션 저장소를 공유하려면 다음 방법 중 하나를 사용하십시오.

- 모든 정책 서버가 세션 저장소 하나를 가리키도록 지정

  정책 서버 관리 콘솔에서 지정된 세션 저장소가 사용되도록 정책 서버를 구성합니다.

- 세션 저장소를 여러 세션 저장소에 복제

  데이터베이스 복제에 대한 지침은 해당 데이터베이스 설명서를 참조하십시오.

제 5 장: 파트너 관계 페더레이션에 대한 사용자 디렉터리 연결

파트너 관계 페더레이션은 사용자 디렉터리의 항목을 조회하여 ID 를 확인하고 지정된 프린서펄에 대한 사용자 특성을 검색합니다. 어설션 당사자에서 페더레이션 파트너는 적절한 사용자에 대한 어설션을 생성하고 각 사용자를 사용자 디렉터리에 대해 인증합니다. 신뢰 당사자에서 페더레이션 파트너는 어설션에서 필요한 정보를 추출하고 사용자 디렉터리에서 적절한 사용자 레코드를 찾습니다.

관리 UI 에서 "인프라", "디렉터리", "사용자 디렉터리"를 선택하여 기존 사용자 디렉터리에 대한 연결을 구성하십시오. 기존 사용자 디렉터리에 대한 연결만 설정하십시오. 새로운 사용자 디렉터리를 구성하는 것이 아닙니다.

참고: 페더레이션된 구성에서 ODBC 데이터베이스를 사용하려면 사용자 디렉터리로 ODBC 데이터베이스를 선택하기 전에 SQL 쿼리 체계와 올바른 SQL 쿼리를 설정하십시오.

필요한 경우 둘 이상의 디렉터리에 대한 연결을 구성하십시오. 디렉터리 유형은 동일하지 않아도 됩니다.

사용자 디렉터리에 대한 자세한 내용은 *정책 서버 구성 안내서*를 참조하십시오.

제 6 장: SiteMinder 세션에서 인증 URL 보호 필요

정책 서버가 어설션을 생성하려면 사용자가 IdP 정책 서버에 세션이 있어야 합니다. 세션을 구성하기 위해 IdP 의 싱글 사인온 서비스가 인증 URL 을 통해 사용자를 응용 프로그램으로 리디렉션합니다. 사용자에게 인증 챌린지가 표시되도록 정책을 사용하여 인증 URL 을 보호하십시오. 그러면 사용자가 로그인하고 세션이 구성됩니다.

인증 URL 이 redirect.jsp 파일을 가리켜야 합니다. 예를 들면 다음과 같습니다.

http://webserver1.example.com/affwebservices/redirectjsp/redirect.jsp

이 예에서 webserver1 은 웹 에이전트 옵션 팩이 있는 웹 서버를 식별합니다. redirect.jsp 파일은 아이덴티티 공급자에 설치된 웹 에이전트 옵션 팩에 포함되어 있습니다.

성공적으로 인증한 후 redirect.jsp 응용 프로그램은 어설션 생성을 위해 사용자를 싱글 사인온 서비스로 다시 리디렉션합니다.

세션을 생성하려면 두 단계가 필요합니다.

1. redirect.jsp 파일에 대한 정책을 만듭니다. (페이지 65)
2. 파트너 관계에 인증 URL 을 지정합니다 (페이지 67).

Redirect.jsp 에 대한 정책 만들기

인증 챌린지를 트리거하기 위해 인증은 인증 URL 을 보호해야 합니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "인프라", "에이전트", "에이전트 만들기"를 차례로 선택합니다.

   어설션 당사자 웹 서버에 대해 정의된 영역에 바인딩하려면 웹 에이전트를 만드십시오. 웹 서버에 대한 고유 에이전트 이름을 할당하십시오.

3. "정책", "도메인", "도메인", "도메인 만들기"를 선택합니다.

   인증 URL 에 대한 정책 도메인을 만듭니다. 챌린지되는 사용자를 포함하는 사용자 디렉터리를 추가하십시오.

4. 정책 도메인에 속한 리소스에 액세스할 수 있어야 하는 사용자를 선택합니다.

5. "영역" 탭을 선택하고 다음 값으로 정책 도메인에 대한 영역을 정의합니다.

   에이전트

   어설션 당사자 웹 서버에 대한 에이전트입니다. 이 에이전트는 2 단계에서 만들었습니다.

   리소스 필터

   /affwebservices/redirectjsp

   이 리소스 필터는 웹 에이전트 및 SPS 페더레이션 게이트웨이에 대해 적용됩니다.

   기본 리소스 보호

   보호됨

   인증 체계

   기본

   영구 세션

   세션 정보를 저장하려면 HTTP-아티팩트 프로필에 대한 영역 대화 상자의 "세션" 섹션에서 "영구" 확인란을 선택하십시오. 세션 정보는 싱글 로그아웃과 같은 기능 및 특성 기관에 필요합니다.

6. 영역 대화 상자의 "규칙" 섹션에서 "규칙 만들기"를 클릭합니다. 필드에 다음 값을 입력합니다.

   리소스

   /*

   별표는 규칙이 영역의 모든 리소스에 적용됨을 의미합니다.

   허용/거부 및 사용/사용 안 함

   액세스 허용

   "사용" 확인란이 선택됩니다.

   작업

   웹 에이전트 작업

   GET, POST, PUT

7. "정책" 탭을 선택하고 다음 구성 요소를 포함하는 정책을 만듭니다.
   - 사용자 디렉터리에서 선택한 사용자 집합
   - redirectjsp 응용 프로그램 및 관련 규칙을 포함하는 영역

이제 정책이 인증 URL 을 보호합니다. 사용자가 이 URL 로 리디렉션되면 인증 챌린지가 트리거됩니다. 이제 세션이 생성됩니다.

파트너 관계에 인증 URL 지정

인증 URL 을 보호하기 위해 정책을 구성한 다음에는 어설션 당사자-신뢰 당사자 파트너 관계(예: IdP->SP 파트너 관계)에 이 URL 을 지정하십시오.

인증 URL 은 싱글 사인온 구성의 일부로서 설정됩니다. 대화 상자의 "인증" 섹션에서 "인증 모드"에 대해 **로컬**을 선택하고 완전한 인증 URL 을 입력하십시오. 예를 들면 다음과 같습니다.

http://webserver1.example.com/affwebservices/redirectjsp/redirect.jsp

이 예에서 webserver1.example.com 은 웹 에이전트 옵션 팩이 있는 웹 서버를 식별합니다.

제 7 장: 페더레이션 엔터티 구성

이 섹션은 다음 항목을 포함하고 있습니다.

엔터티를 생성하는 방법

페더레이션 파트너 관계의 각 파트너는 *페더레이션 엔터티*로 간주됩니다. 파트너 관계를 설정하기 전에 로컬 파트너를 나타내는 로컬 엔터티와 원격 파트너를 나타내는 원격 엔터티를 정의하십시오.

페더레이션 엔터티를 구성하는 두 가지 방법은 다음과 같습니다.

- 메타데이터를 사용하지 않고 엔터티를 생성합니다 (페이지 69).
- 메타데이터를 가져와서 엔터티를 생성합니다 (페이지 75).

메타데이터를 사용하지 않고 엔터티 만들기

메타데이터 없이 엔터티를 생성하려면 다음 프로세스를 사용하십시오.

1. 엔터티 유형을 지정합니다.
2. 엔터티 유형에 대한 구체적 사항을 구성합니다.
3. 엔터티 구성을 확인합니다.

엔터티 유형 선택

엔터티를 구성하는 첫 번째 단계는 엔터티 유형을 설정하고 엔터티 역할을 결정하는 것입니다.

엔터티 유형을 설정하려면

1. 관리 UI 에 로그인합니다.
2. "페더레이션", "파트너 관계 페더레이션", "엔터티"를 선택합니다.

3. "엔터티 만들기"를 클릭합니다.

   "엔터티 만들기" 대화 상자가 표시됩니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

4. 다음 옵션 중 *하나*를 선택하십시오.

   로컬

   사이트에 대하여 로컬인 엔터티를 생성하는 것임을 나타냅니다.

   원격

   원격 사이트에서 파트너를 나타내는 엔터티를 구성하는 것임을 나타냅니다.

5. 나머지 필드를 구성합니다.

   새 엔터티 유형

   어설션 또는 신뢰 당사자를 선택합니다.

   SAML 토큰 유형(WS-FED 만 해당)

   사용자 자격 증명 정보를 포함하는 암호화된 토큰의 SAML 형식을 정의하는 토큰 유형을 선택합니다. 토큰이 WS-페더레이션 1.0 에 대한 SAML 토큰 유형을 준수하도록 하려는 경우에만 "레거시" 옵션을 선택합니다.

6. "다음"을 클릭하여 엔터티의 세부 사항을 구성합니다.

상세한 로컬 엔터티 구성

엔터티 유형을 지정한 후에는 엔터티의 상세 정보를 구성하십시오. 로컬 엔터티의 경우 다음 정보를 정의하십시오.

- 엔터티에 대한 ID 정보
- 서명 및 암호화 옵션
- 이름 ID 형식 및 특성

다음 단계를 수행하십시오.

1. "엔터티 구성" 단계부터 시작합니다.
2. 구성 중인 로컬 엔터티 유형의 기능 및 서비스에 대한 모든 필수 필드에 데이터를 입력합니다.

   필드의 설명을 보려면 "도움말"을 클릭하십시오.
3. "다음"을 클릭합니다.

   "확인" 대화 상자가 표시됩니다.

다음 사항에 유의하십시오.

엔터티 ID 및 엔터티 이름 설정

엔터티 ID 가 원격 파트너를 나타내는 경우 이 값은 고유해야 합니다. 엔터티 ID 가 로컬 파트너를 나타내는 경우에는 동일한 시스템에서 재사용될 수 있습니다.

엔터티 이름은 정책 저장소의 엔터티 개체를 식별합니다. 엔터티 이름은 고유한 값이어야 합니다. 이 값은 내부용으로만 사용되고 원격 파트너는 이 값을 알지 못합니다.

참고: 엔터티 이름은 엔터티 ID 와 값이 동일할 수 있지만 동일한 사이트의 다른 엔터티와 값을 공유하지는 않습니다.

서명 및 암호화 기능

서명 및 암호화 기능을 사용하려면 인증서 데이터 저장소에 해당 키/인증서 항목이 있어야 합니다. 해당 키/인증서 항목이 없을 경우 "가져오기"를 클릭하여 로컬 시스템의 파일에서 개인 키/인증서 쌍을 가져오십시오. 트러스트된 인증서를 가져올 수도 있습니다.

참고: SAML 2.0 POST 프로필을 사용하는 경우 서명 어설션이 필요합니다.

WSFED 특성(WS-페더레이션만 해당)

WS-페더레이션 엔터티가 통신할 수 있도록 다양한 서비스 URL 및 ID 를 지정할 수 있습니다.

이름 ID 형식

페더레이션된 엔터티가 지원하는 식별자 유형을 지정할 수 있습니다.

어설션 특성 구성(어설션 파트너만 해당)

어설션을 생성할 때 구체적인 어설션 특성을 포함하도록 어설션 당사자를 구성할 수 있습니다. 이러한 특성은 엔터티 수준에서 정의하는 것이 좋습니다. 이 엔터티는 파트너 관계를 위한 템플릿 역할을 하므로 엔터티에 대해 정의하는 모든 어설션 특성이 해당 파트너 관계에 전파됩니다. 엔터티 수준에서 어설션 특성을 정의하면 여러 파트너 관계에서 엔터티를 사용할 수 있는 이점이 있습니다.

파트너 관계에 대한 어설션 특성을 추가하거나 제거하려면 엔터티 수준이 아닌 파트너 관계 수준에서 이러한 수정을 하십시오.

상세한 원격 엔터티 구성

엔터티 유형을 지정한 후에는 엔터티의 상세 정보를 구성하십시오. 원격 엔터티 유형의 경우 다음 정보를 정의하십시오.

- 엔터티에 대한 ID 정보
- 서명 및 암호화 옵션
- NameID 및 특성 정보

다음 단계를 수행하십시오.

1. "엔터티 구성" 단계부터 시작합니다.
2. 어설션 소비자 서비스 URL 을 지정합니다. 예:
   - SP 가 Google 같은 사이트인 경우 URL 을 다음과 같이 지정할 수 있습니다.

     https://www.google.com/a/example.com/acs
   - SP 가 Salesforce.com 같은 사이트인 경우 URL 을 다음과 같이 지정할 수 있습니다.

     https://login.salesforce.com/?saml=EK05LGnm40H7
   - SP 가 다른 비즈니스 파트너인 경우 URL 은 다음과 유사할 수 있습니다.

     http://myserver.forwardinc.com:9080/samlsp/acs
3. 원격 엔터티 유형의 기능 및 서비스에 대한 다른 모든 필수 필드에 데이터를 입력합니다.

   필드 설명을 보려면 "도움말"을 클릭하십시오.
4. "다음"을 클릭합니다.

   "확인" 대화 상자가 표시됩니다.

다음 사항에 유의하십시오.

엔터티 ID 및 엔터티 이름 설정

엔터티 ID 가 원격 파트너를 나타내는 경우 이 값은 고유해야 합니다. 엔터티 ID 가 로컬 파트너를 나타내는 경우에는 동일한 시스템에서 재사용될 수 있습니다.

엔터티 이름은 정책 저장소의 엔터티 개체를 식별합니다. 엔터티 이름은 고유한 값이어야 합니다. 이 값은 내부용으로만 사용되고 원격 파트너는 이 값을 알지 못합니다.

참고: 엔터티 이름은 엔터티 ID 와 값이 동일할 수 있지만 동일한 사이트의 다른 엔터티와 값을 공유하지는 않습니다.

서명 및 암호화 기능

서명 및 암호화 기능을 사용하려면 인증서 데이터 저장소에 해당 키/인증서 항목이 있어야 합니다. 해당 키/인증서 항목이 없을 경우 "가져오기"를 클릭하여 로컬 시스템의 파일에서 개인 키/인증서 쌍을 가져오십시오. 트러스트된 인증서를 가져올 수도 있습니다.

참고: SAML 2.0 POST 프로필을 사용하는 경우 서명 어설션이 필요합니다.

WSFED 특성(WS-페더레이션만 해당)

WS-페더레이션 엔터티가 통신할 수 있도록 다양한 서비스 URL 및 ID 를 지정할 수 있습니다.

이름 ID 형식

페더레이션된 엔터티가 지원하는 식별자 유형을 지정할 수 있습니다.

어설션 특성 구성(어설션 파트너만 해당)

어설션을 생성할 때 구체적인 어설션 특성을 포함하도록 어설션 당사자를 구성할 수 있습니다. 이러한 특성은 엔터티 수준에서 정의하는 것이 좋습니다. 이 엔터티는 파트너 관계를 위한 템플릿 역할을 하므로 엔터티에 대해 정의하는 모든 어설션 특성이 해당 파트너 관계에 전파됩니다. 엔터티 수준에서 어설션 특성을 정의하면 여러 파트너 관계에서 엔터티를 사용할 수 있는 이점이 있습니다.

파트너 관계에 대한 어설션 특성을 추가하거나 제거하려면 엔터티 수준이 아닌 파트너 관계 수준에서 이러한 수정을 하십시오.

엔터티 구성 확인

엔터티 구성을 저장하기 전에 검토하십시오.

다음 단계를 수행하십시오.

1. 엔터티 대화 상자에서 설정을 검토합니다.
2. "뒤로"를 클릭하여 이 대화 상자의 모든 설정을 수정합니다.
3. 원하는 대로 구성되었으면 "마침"을 클릭합니다.

이제 새 엔터티가 구성되었습니다.

파트너 관계에서 엔터티 구성 변경

단일 파트너 관계 구성의 컨텍스트 내에서 원격 엔터티에 대한 엔터티 ID 값을 변경할 수 있습니다. 하지만 파트너 관계 수준에서 엔터티 ID 를 변경하더라도 파트너 관계가 다른 엔터티로 연결되지 않으며 원본 엔터티가 업데이트되지도 않습니다. 엔터티 수정은 엔터티에서 파트너 관계로의 단방향 전파입니다. 파트너 관계 수준에서 엔터티 ID 에 대한 변경 내용은 원본 엔터티로 전파되지 않습니다.

참고: 지정하는 엔터티 ID 는 원격 파트너가 사용하는 ID 와 일치해야 합니다.

엔터티 구성을 템플릿이라고 생각하십시오. 파트너 관계는 엔터티 템플릿을 기반으로 생성되므로 파트너 관계를 변경하더라도 원본 엔터티 템플릿은 변경되지 않습니다.

파트너 관계 내의 엔터티에 대한 자세한 내용은 파트너 관계의 엔터티 편집 (페이지 84)을 참조하십시오.

메타데이터를 가져와서 엔터티 만들기

메타데이터 파일에서 데이터를 가져와서 페더레이션 엔터티를 생성할 수 있습니다. 메타데이터를 가져오면 파트너 관계를 생성하기 위한 구성의 양이 줄어듭니다.

메타데이터는 다음과 같은 방법으로 이용할 수 있습니다.

- 원격 파트너에서 데이터를 가져와서 새 원격 엔터티를 생성합니다.
- 원격 파트너에서 데이터를 가져와서 기존 원격 엔터티를 업데이트합니다.
- 로컬 엔터티에서 데이터를 가져와서 새 로컬 엔터티를 생성합니다.

  이 옵션은 다른 페더레이션 제품에서 마이그레이션할 때 유용합니다.

  참고: 페더레이션에서는 기존 파트너 관계 및 로컬 엔터티를 업데이트하거나 복원하기 위한 메타데이터 가져오기를 지원하지 않습니다. 기존 로컬 엔터티를 업데이트하려면 엔터티를 편집하여 변경할 설정을 수정하십시오. 메타데이터는 *새* 로컬 엔터티를 생성하는 용도로만 가져올 수 있습니다.

메타데이터 기반 엔터티를 생성하는 프로세스는 다음과 같습니다.

1. 새 엔터티를 구성하기 위한 메타데이터 파일을 선택합니다.
2. 메타데이터 파일에서 엔터티 항목을 선택합니다. 파일에는 여러 개의 엔터티가 포함될 수 있지만 파일당 엔터티가 하나인 것이 좋습니다.
3. (선택 사항) 인증서 데이터 저장소로 가져올 인증서를 선택합니다. 인증서는 메타데이터 파일에 있어야 합니다.

   이 인증서는 인증서 요청 확인, 싱글 로그아웃 응답 확인(SAML 2.0) 및 암호화(SAML 2.0)에 사용할 수 있습니다.
4. 엔터티 구성을 확인합니다.

이 단계에 대한 상세 정보는 다음 섹션에서 설명합니다.

메타데이터 파일 선택

메타데이터를 사용하여 항목을 생성하는 첫 번째 단계는 메타데이터 파일을 선택하는 것입니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "페더레이션", "파트너 관계 페더레이션", "엔터티"를 선택합니다.
3. "메타데이터 가져오기"를 클릭합니다.

   "메타데이터 가져오기" 대화 상자가 열립니다.

   필드 설명을 보려면 "도움말"을 클릭하십시오.
4. 엔터티 만들기에 사용할 메타데이터 파일을 찾습니다.

5. 새 로컬 또는 원격 엔터티를 생성할지 아니면 기존의 원격 엔터티를 업데이트할지 선택합니다.

   참고: 정책 서버는 파트너 관계 및 로컬 엔터티를 업데이트하기 위해 메타데이터를 가져오는 것을 지원하지 않습니다. 새 로컬 엔터티만 생성할 수 있습니다. 기존 로컬 엔터티를 업데이트하려면 엔터티를 편집하여 변경할 설정을 수정하십시오. 기존 원격 엔터티를 업데이트하거나 새 원격 엔터티를 생성할 수 있습니다.

6. "다음"을 클릭하여 파일에서 엔터티를 선택합니다.

만료된 항목이 있는 메타데이터 파일을 선택하면 UI 에서 표시하는 다음 대화 상자에 만료된 항목이 나열된 섹션이 포함됩니다. 이 만료된 항목은 선택할 수 없으며 참조용으로만 표시됩니다. 메타데이터 파일의 모든 엔터티가 만료되면 엔터티가 표시되지 않습니다. 이 경우 새 문서를 업로드하십시오.

가져올 엔터티 선택

이 절차에서는 엔터티를 생성할 메타데이터 파일을 이미 선택한 것으로 가정합니다. 파일에서 엔터티를 선택하십시오.

다음 단계를 수행하십시오.

1. "파일에 정의된 엔터티 선택" 대화 상자에서 새 항목의 이름을 지정합니다.

   엔터티를 생성하기 위해 로컬 가져오기를 수행하는 경우에는 파트너 관계 이름을 정의합니다.

2. 옵션 버튼을 클릭하여 엔터티를 선택합니다.

3. "다음"을 클릭합니다.

   원격 엔터티의 메타데이터를 가져오는 경우 문서에 인증서 데이터가 포함되어 있으면 "인증서 가져오기" 대화 상자가 표시됩니다.

   가져온 메타데이터 파일에 인증서 항목이 포함된 경우 이러한 항목을 가져올 수 있습니다.

인증서 가져오기

서명된 어설션을 확인하려면 메타데이터에 인증서가 포함된 경우 인증서를 가져오십시오. 메타데이터가 인증서가 포함되지 않은 경우 이 단계를 건너뛰고 확인 단계로 이동하십시오.

다음 단계를 수행하십시오.

1. "인증서 가져오기" 단계에서 가져올 메타데이터 파일의 인증서 항목을 선택합니다.

   올바르지 않은 항목이 있는 인증서 파일을 선택하면 다음 대화 상자에 만료된 항목이 나열된 섹션이 포함됩니다. 이러한 만료된 항목은 선택할 수 없습니다. 이러한 항목은 참조용으로만 표시됩니다. 파일의 모든 항목이 올바르지 않은 경우에는 가져오기 마법사가 인증서 선택 단계를 건너뜁니다.

   선택한 각 항목에 고유한 별칭을 지정합니다.

2. "다음"을 클릭합니다.

   항목 테이블을 보여 주는 "확인" 대화 상자가 표시됩니다.

   메타데이터 파일에서 인증서가 동일한 두 항목을 선택할 수 있습니다. SAML 1.1 및 WS-페더레이션 메타데이터의 경우 SAML 1.1 은 데이터를 암호화하지 않으므로 모든 항목의 인증서 용도가 "서명"으로 표시됩니다.

   SAML 2.0 의 경우 각 항목의 용도가 인증서마다 다른 용도로(예를 들어 하나는 서명용, 다른 하나는 암호용) 표시될 수 있습니다. 확인 단계로 이동하면 창에 단일 인증서 항목이 있는 테이블이 표시됩니다. 인증서 용도는 "서명 및 암호화"로 나열됩니다. 이 항목은 이전에 선택한 두 항목의 조합입니다. 이 항목은 선택한 인증서 항목에 대해 지정된 첫 번째 별칭도 사용합니다.

   이러한 상황은 두 용도 모두에 대해 동일한 인증서가 메타데이터 파일에 나열된 경우에만 발생합니다. 파일에 별도의 인증서 두 개가 포함된 경우 확인 단계에서 테이블에 두 항목이 모두 표시됩니다.

예를 들어 메타데이터 파일에서 두 항목을 선택했는데 두 항목이 동일한 인증서라는 사실을 모르는 경우가 있습니다. 첫 번째 용도는 "서명"이며 이를 별칭 **cert1** 에 할당합니다.두 번째 용도는 "암호화"이며 이를 별칭 **cert2** 에 할당합니다. 가져오기를 확인하면 "선택된 인증서 데이터"라는 제목의 테이블에 다음과 유사한 항목이 표시됩니다.

별칭	발급된 대상	사용
cert1	Jane Doe	서명 및 암호화

메타데이터 파일에 용도가 지정되지 않은 경우 용도가 기본값 "서명 및 암호화"로 설정됩니다.

3. "다음"을 클릭하여 구성을 마칩니다.

엔터티 구성 확인

엔터티 구성을 저장하기 전에 검토하십시오.

다음 단계를 수행하십시오.

1. 엔터티 대화 상자에서 설정을 검토합니다.
2. "뒤로"를 클릭하여 이 대화 상자의 모든 설정을 수정합니다.
3. 원하는 대로 구성되었으면 "마침"을 클릭합니다.

이제 새 엔터티가 구성되었습니다.

제 8 장: 파트너 관계 생성 및 활성화

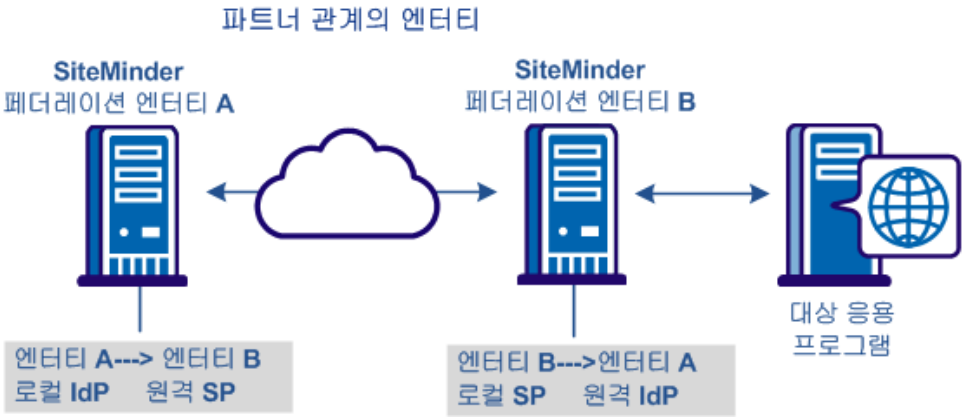
이 섹션은 다음 항목을 포함하고 있습니다.

파트너 관계 생성

파트너 관계 페더레이션의 기본 목적은 두 조직이 사용자 아이덴티티 정보를 공유하고 SSO(싱글 사인온)를 사용할 수 있도록 조직 간에 파트너 관계를 설정하는 것입니다. 파트너 관계는 서로 다른 사이트에 있는 두 개의 엔터티로(로컬과 원격에 각각 하나씩) 구성됩니다. 각 엔터티는 어설션 당사자, 어설션 또는 신뢰 당사자를 생산하는 측, 어설션을 소비하는 측의 역할을 수행할 수 있습니다.

SiteMinder 가 두 사이트에 모두 설치된 경우 각 사이트마다 파트너 관계를 정의해야 합니다. 한 사이트의 각 로컬 어설션 당사자-신뢰 당사자 파트너 관계마다, 파트너 사이트에 이에 대응하는 로컬 신뢰 당사자-어설션 당사자 파트너 관계가 있어야 합니다. 예를 들어, 엔터티 A 에서 파트너 관계 구성의 경우 엔터티 A 는 로컬 아이덴티티 공급자(IdP)이고 엔터티 B 는 원격 서비스 공급자(SP)입니다. 엔터티 B 에서 파트너 관계 구성의 경우 엔터티 B 는 로컬 서비스 공급자(SP)이고 엔터티 A 는 원격 아이덴티티 공급자(IdP)입니다. 관점은 로컬 엔터티에 기초합니다.

다음 그림은 파트너 관계에 대한 엔터티 관계를 보여 줍니다.



참고: 어설션 당사자는 둘 이상의 신뢰 당사자와 파트너 관계를 가질 수 있으며 신뢰 당사자는 둘 이상의 어설션 당사자와 파트너 관계를 설정할 수 있습니다.

파트너 관계를 만들려면 파트너 관계 마법사가 필요한 구성 단계를 안내합니다.

파트너 관계 정의

페더레이션 파트너 관계 정의는 어떤 페더레이션 파트너가 로컬이고 어떤 페더레이션 파트너가 원격인지를 지정합니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.

   "페더레이션 파트너 관계" 대화 상자가 표시됩니다.
3. "페더레이션 파트너 관계" 목록에서 "파트너 관계 만들기"를 클릭합니다.

4. 다음 파트너 관계 중 하나를 선택하십시오.
   - SAML2 IDP->SP(아이덴티티 공급자가 로컬)
   - SAML2 SP->IDP(서비스 공급자가 로컬)
   - SAML1.1 생산자->소비자(생산자가 로컬)
   - SAML1.1 소비자->생산자(소비자가 로컬)
   - WSFED IP->RP(아이덴티티 공급자가 로컬)
   - WSFED RP->IP(리소스 파트너가 로컬)

파트너 관계 마법사의 첫 번째 단계에서 파트너 관계 대화 상자가 열립니다.

파트너 관계 식별 및 구성

마법사의 "파트너 관계 구성" 단계에서 파트너 관계의 이름을 지정하고 로컬 및 원격 엔터티를 지정하여 파트너 관계를 식별하십시오.

참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

다음 단계를 수행하십시오.

1. 파트너 관계의 이름을 입력합니다. 이름에는 영숫자 문자, 밑줄, 하이픈 및 마침표를 사용할 수 있습니다. 공백은 사용할 수 없습니다.
2. (선택 사항) 설명을 입력합니다.
3. 엔터티를 이미 구성한 경우 로컬 목록에서 로컬 엔터티를 선택합니다. 그렇지 않은 경우에는 "로컬 엔터티 만들기"를 클릭합니다.
4. 엔터티를 이미 구성한 경우 원격 목록에서 원격 엔터티를 선택합니다. 그렇지 않은 경우에는 "원격 엔터티 만들기"를 클릭합니다.

   참고: 나중에 메타데이터를 가져와서 원격 엔터티를 만들 계획인 경우 이 단계를 미룰 수 있습니다.
5. (선택 사항) 기준 URL 을 지정합니다.

6. (선택 사항) "차이 시간"을 초 단위로 입력합니다.

   차이 시간은 로컬 시스템의 시스템 시간과 원격 시스템의 시스템 시간 사이의 차이입니다. 일반적으로 시스템 시계가 부정확할 때 이런 상태가 발생합니다. 현재 시간에서 초를 빼서 차이 시간을 결정하십시오.

   시스템에서는 차이 시간 및 SSO 유효 기간을 사용하여 어설션의 유효 기간을 결정합니다.

7. "사용 가능한 디렉터리" 목록에서 하나 이상의 사용자 디렉터리를 선택하고 이를 "선택된 디렉터리" 목록으로 이동합니다.

   사용자 디렉터리를 하나만 구성한 경우에는 이 디렉터리가 자동으로 "선택된 디렉터리" 목록으로 들어갑니다.

   중요! 사용자 디렉터리로 ODBC 데이터베이스를 사용하려면 SQL 쿼리 체계 및 올바른 SQL 쿼리를 정의하십시오. 사용자 디렉터리로 선택하려면 다음 단계를 수행해야 합니다.

8. "다음"을 클릭하여 파트너 관계 마법사를 계속합니다. 마법사의 단계를 따라 파트너 관계의 여러 기능(일부는 필수, 일부는 선택적 기능)을 구성할 수 있습니다. 이러한 기능의 구성 정보는 이 안내서의 이후 단원에서 설명됩니다.

참고: 파트너 관계를 편집하는 경우에는 이 필드 옆의 "업데이트 가져오기"를 클릭하여 엔터티 정보를 업데이트할 수 있습니다. 엔터티 구성의 최신 정보가 파트너 관계에 전파됩니다. 하지만 엔터티 정보를 파트너 관계에서 직접 편집하는 경우에는 변경 내용이 다시 개별 엔터티 구성으로 전파되지 않습니다.

파트너 관계의 엔터티 편집

로컬 및 원격 엔터티 필드 옆의 "업데이트 가져오기"를 클릭하여 엔터티에 대한 정보를 업데이트할 수 있습니다. "업데이트 가져오기"를 선택하면 시스템은 엔터티에서 최신 정보를 가져올지 묻습니다.

확인 후에는 편집하는 파트너 관계가 최신 엔터티 정보로 새로 고쳐집니다. 파트너 관계 마법사를 완료하면 변경 내용이 저장됩니다. 업데이트를 확인하지 않으면 파트너 관계 구성이 동일하게 유지됩니다.

"엔터티 이름"은 정책 저장소의 엔터티 개체를 식별합니다. 제품에서는 이 값을 사용하여 엔터티를 내부적으로 구분하기 때문에 "엔터티 이름"은 고유 식별자여야 합니다. 이 값은 외부에서 사용되지 않으며 원격 파트너는 이 값을 알지 못합니다.

엔터티 ID 가 원격 파트너를 나타내는 경우 이 값은 고유해야 합니다. 엔터티 ID 가 로컬 파트너를 나타내는 경우에는 동일한 시스템에서 재사용될 수 있습니다.

참고: "엔터티 이름"의 값은 "엔터티 ID"와 동일할 수 있지만 값을 다른 엔터티와 공유하지 마십시오.

엔터티는 페더레이션 파트너 관계의 핵심 구성 요소입니다. 엔터티를 변경하면 파트너 관계가 크게 변경됩니다. 따라서 관리 UI 에서는 파트너 관계가 설정된 후 엔터티를 바꿀 수 없습니다. 엔터티를 바꾸려면 파트너 관계를 생성하십시오.

엔터티 ID 는 엔터티를 고유하게 식별하지 않기 때문에 파트너 관계 구성 내에서 약간의 유연성을 제공하기 위해 엔터티 ID 를 변경할 수 있습니다. 파트너 관계 수준에서 엔터티 ID 를 변경하더라도 파트너 관계가 다른 엔터티로 연결되지 않습니다. 파트너 관계의 원래 엔터티는 변경되지 않습니다. 엔터티 수정은 엔터티에서 파트너 관계로의 단방향 전파입니다. 파트너 관계의 엔터티 ID 변경 내용은 원본 엔터티로 다시 전파되지 않습니다.

엔터티 구성을 템플릿이라고 생각하십시오. 파트너 관계는 엔터티 템플릿을 기반으로 생성되므로 파트너 관계를 변경하더라도 원본 엔터티 템플릿은 변경되지 않습니다.

파트너 관계 확인

파트너 관계 구성을 저장하기 전에 검토하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 "확인" 단계에서 설정을 검토합니다.
2. 각 그룹 상자에서 "수정"을 클릭하여 설정을 변경합니다.
3. 원하는 대로 구성되었으면 "마침"을 클릭합니다.

파트너 관계 구성이 완료되었습니다.

파트너 관계 활성화

파트너 관계에 필요한 모든 설정을 구성한 후에는 파트너 관계를 사용할 수 있도록 활성화하십시오. 동일한 프로세스를 사용하여 파트너 관계를 비활성화할 수도 있습니다.

다음 단계를 수행하십시오.

1. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.

   "파트너 관계" 대화 상자가 열립니다.

2. "작업" 메뉴에서 원하는 파트너 관계 옆의 "활성화" 또는 "비활성화"를 선택합니다.

   확인 대화 상자가 표시됩니다.

   참고: 활성화는 상태가 "정의됨" 또는 "비활성"인 파트너 관계에만 사용할 수 있습니다. 비활성화는 상태가 "활성화"인 파트너 관계에만 사용할 수 있습니다.

3. "예"를 클릭하여 선택을 확인합니다.

   파트너 관계의 상태가 설정되고 표시가 새로 고쳐집니다.

중요! 파트너 관계를 수정하기 전에는 비활성화하십시오.

파트너 관계 내보내기

메타데이터를 원격 엔터티를 생성하고 파트너 관계를 형성하는 기반으로 사용할 수 있습니다. 메타데이터를 사용하면 엔터티의 많은 측면이 이미 메타데이터 파일에 정의되어 있으므로 파트너 관계를 더 효율적으로 구성할 수 있습니다. 그런 다음 파일을 가져와서 파트너 관계 또는 원격 엔터티를 생성할 수 있습니다.

파트너 관계를 완료해야만 내보낼 수 있는 것은 아닙니다. 파트너 관계의 일부만 구성한 후 내보낼 수 있습니다.

관리 UI 에서는 기존 파트너 관계 항목의 메타데이터를 내보낼 수 있습니다.

참고: 관리 UI 에서는 기존 로컬 어설션 또는 신뢰 엔터티의 메타데이터를 내보낼 수 있습니다. SAML 1.1 데이터를 내보낼 때 결과 메타데이터 파일에 사용되는 용어는 SAML 2.0 용어입니다. 이러한 규칙은 SAML 사양의 일부입니다. SAML 1.1 데이터를 가져올 때 용어는 SAML 1.1 용어를 사용하여 올바르게 가져오게 됩니다.

파트너 관계에서 내보낼 때는 선택된 파트너 관계가 내보내기의 기반으로 사용됩니다. 새 파트너 관계 이름은 정의할 수 없습니다. SiteMinder 는 선택된 파트너 관계의 이름을 사용합니다.

다음 단계를 수행하십시오.

1. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.

   "파트너 관계" 대화 상자가 표시됩니다.

2. 목록에서 적절한 항목 옆의 "작업" 풀다운 메뉴를 클릭하고 "메타데이터 내보내기"를 선택합니다.

   "메타데이터 내보내기" 대화 상자가 열립니다.

3. 대화 상자의 필드를 입력합니다.

   상태가 "활성화"인 파트너 관계를 내보낼 때는 대부분의 필드가 읽기 전용입니다. "유효 기간" 필드와 별칭 드롭다운 목록만 수정할 수 있습니다.

4. "내보내기"를 클릭하여 완료합니다.

5. 메타데이터 파일을 열지 아니면 저장할지 묻는 대화 상자가 표시됩니다. 파일을 열어서 볼 수 있습니다.

6. 데이터를 로컬 시스템의 XML 파일에 저장합니다.

메타데이터가 지정된 XML 파일로 내보내졌습니다.

제 9 장: 파트너 관계에 대한 페더레이션된 사용자 식별

이 섹션은 다음 항목을 포함하고 있습니다.

어설션 당사자에서의 페더레이션 사용자 구성

로컬 엔터티가 어설션 당사자인 경우 파트너 관계 마법사의 두 번째 단계에서는 "페더레이션 사용자" 대화 상자가 나타납니다. 이 단계에서는 원격 사이트의 대상 리소스에 액세스할 수 있는 권한을 부여할 사용자를 지정할 수 있습니다.

다음 단계를 수행하십시오.

참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

1. "페더레이션된 사용자" 그룹 상자의 표에 있는 "디렉터리" 열의 목록에서 사용자 디렉터리를 선택합니다.

   이 풀다운 목록은 이전 대화 상자에서 지정한 디렉터리 수에 따라 한 개 이상의 디렉터리 항목으로 구성됩니다.

2. "사용자 클래스" 열에서 사용자 클래스를 선택합니다. 이 항목은 인증할 수 있는 개별 사용자 또는 사용자 그룹의 범주를 지정합니다. 이 필드의 옵션은 사용자 디렉터리(LDAP 또는 ODBC)의 유형에 따라 다릅니다. 각 사용자 클래스의 설명 및 예제는 "사용자 클래스" 표를 참조하십시오.

3. 사용자 이름/필터 기준 열에 이름 또는 필터를 입력합니다. 이 열의 값을 사용하여 시스템은 페더레이션된 사용자를 인증하는 사용자 또는 사용자 그룹을 찾을 수 있습니다. 이 항목은 "사용자 클래스" 열에 대해 선택하는 값에 따라 다릅니다. 이름 및 필터의 예제는 이 절차의 끝에 나오는 표를 참조하십시오.

4. (선택 사항) 특정 항목에 대해 "제외"를 선택하여 해당 사용자 클래스를 제외하도록 지정할 수 있습니다. 기본값은 디렉터리의 모든 사용자를 포함하는 것입니다.

   참고: 제외 조건과 포함 조건이 충돌할 경우 항상 제외 조건이 포함 조건보다 우선합니다.

5. (선택 사항) 동일한 디렉터리 또는 다른 사용자 디렉터리에 대해 다른 사용자 클래스를 지정하려면 "행 추가"를 클릭합니다.

사용자 선택이 완료되었습니다.

사용자 클래스 항목의 예제

LDAP 예제

항목을 지정할 때 LDAP 필터 구문을 사용하십시오.

사용자 클래스	유효한 항목
사용자	사용자의 고유 이름입니다. 예: uid=user1,ou=People,dc=example,dc=com
그룹	목록에서 선택된 그룹입니다. 예: ou=Sales,dc=example,dc=com
조직 단위	목록에서 선택된 조직 단위입니다. 예: ou=People,dc=example,dc=com
사용자 속성 필터링	LDAP 필터입니다. 현재 사용자부터 검색이 시작됩니다. **예 1:** mail=user@example.com **예 2:** (|(mail=*@.example.com)(memberOf=cn=Employees, ou=Groups,dc=example,dc=com))

사용자 클래스	유효한 항목
그룹 속성 필터링	LDAP 필터입니다. 현재 사용자가 필터와 일치하는 그룹 중 하나의 구성원인 경우 현재 사용자가 권한 부여됩니다. SiteMinder 레지스트리에 구성된 대로 그룹의 objectclass 가 필터와 결합됩니다. **예 1:** 비즈니스 범주가 "CA Support"인 그룹의 구성원인 사용자를 권한 부여하려면 businessCategory=CA Support 를 입력하십시오. **예 2:** 설명에 "Administrator"가 포함되어 있고 비즈니스 범주가 "Administration"인 그룹의 구성원인 사용자를 권한 부여하려면 (|(description=*Administrator*)(businessCategory=Administration))를 입력하십시오. **참고:** 그룹의 일부 특성은 검색 조건으로 사용되지 않습니다.
OU 속성 필터링	LDAP 필터입니다. 현재 사용자가 필터와 일치하는 조직 단위에 속하는 경우 현재 사용자가 권한 부여됩니다. SiteMinder 레지스트리에 구성된 대로 조직 단위의 objectclass 가 필터와 결합됩니다. **예 1:** 우편 주소가 "12345"인 조직 단위 내 사용자를 권한 부여하려면 postalCode=12345 를 입력하십시오. **예 2:** 기본 설정 배송 방법이 "phone"으로 끝나고 지역이 "London"인 조직 단위의 사용자를 권한 부여하려면 (|(preferredDeliveryMethod=*phone)(l=London))를 입력하십시오.
모두 필터링	LDAP 필터입니다. 현재 사용자가 필터와 일치하는 경우 현재 사용자가 권한 부여됩니다. **예 1:** 부서가 "CA Support"인 사용자를 권한 부여하려면 department=CA Support 를 입력하십시오. **예 2:** "Administrators" 그룹의 구성원이고 부서 번호가 "123" 또는 "789"인 사용자를 권한 부여하려면 (&(memberof=cn=Administrators,ou=Groups,dc=example,dc=com)(|(departmentNumber=123)(departmentNumber=789)))를 입력하십시오.

ODBC 예제

쿼리를 지정할 때 SQL 구문을 사용하십시오.

사용자 클래스	유효한 항목
사용자	사용자에 대한 "이름" 열의 값입니다. 현재 사용자가 항목과 일치하는 경우 현재 사용자가 권한 부여됩니다. 예: user1
그룹	사용자 그룹의 "이름" 열의 값입니다. 현재 사용자가 쿼리와 일치하는 그룹의 구성원인 경우 현재 사용자가 권한 부여됩니다. 예: Administrators
쿼리	SQL SELECT 문입니다. 현재 사용자가 쿼리와 일치하는 경우 현재 사용자가 권한 부여됩니다. **예 1:** user1 의 userid: 입력: SELECT * FROM SmUser 결과 쿼리: SELECT * FROM SmUser WHERE Name = 'user1' **예 2:** user1 의 userid: 입력: SELECT * FROM SmUser WHERE Status LIKE 'Active%' 결과 쿼리: SELECT * FROM SmUser WHERE Status LIKE 'Active%' AND Name = 'user1' **예 3:** user1 의 userid: 입력: SELECT * FROM SmUser WHERE Location IN ('London', 'Paris') 결과 쿼리: SELECT * FROM SmUser WHERE Location IN ('London', 'Paris') AND Name = 'user1'

신뢰 당사자 측에서의 사용자 식별

신뢰 당사자 측에서 파트너는 로컬 사용자 디렉터리의 사용자를 찾을 수 있어야 합니다. 사용자 디렉터리에서 사용자를 찾는 과정이 바로 명확성 프로세스입니다. "사용자 ID" 대화 상자에서 사용자 명확성을 위한 아이덴티티 특성을 구성하십시오.

정책 서버는 명확성 프로세스를 위해 다음 방법 중 하나를 사용합니다.

- 어설션에서 "이름 ID" 값을 추출합니다.
- 어설션의 특정한 특성 값을 사용합니다.
- Xpath 쿼리가 가져오는 값을 사용합니다.

  Xpath 쿼리는 어설션에서 "이름 ID" 이외의 특성을 찾아서 추출합니다.

어설션에서 추출되는 특성을 확인한 후에 이 특성을 검색 사양에 포함하십시오. 명확성 프로세스에 성공하면 정책 서버는 사용자에 대한 세션을 생성합니다.

SAML 2.0 의 경우 어설션 당사자가 사용자 식별자를 생성할 수 있도록 허용하는 AllowCreate 기능 (페이지 94)을 구성할 수도 있습니다.

신뢰 당사자 측에서 사용자 ID 구성

신뢰 당사자가 로컬 사용자 디렉터리에서 사용자를 찾을 수 있도록 사용자 ID 를 구성해야 합니다.

다음 단계를 수행하십시오.

1. 다음의 명확성 특성 중 하나를 선택하십시오.
   - 이름 ID
   - 이전에 채워진 드롭다운 목록의 특성

     원격 어설션 특성이 특성을 포함한 메타데이터에 기반하여 생성된 경우 목록이 채워집니다.
   - 사용자가 입력하는 특성

     이 옵션은 메타데이터를 사용할 수 없고 원격 어설션 엔터티에 특성이 포함되지 않은 경우에 사용될 가능성이 많습니다.
   - Xpath 쿼리

   필드 설명을 보려면 "도움말"을 클릭하십시오.
2. (선택 사항 - SAML 2.0 만) "IDP 가 사용자 식별자를 만들도록 허용"을 선택합니다.

   이 특성은 어설션 당사자 측에서 이 기능을 사용할 수 있는 경우 어설션 당사자가 NameID 에 대한 새 값을 생성하도록 지시합니다. 어설션 당사자의 "이름 ID 형식" 항목은 영구 식별자여야 합니다.

3. (선택 사항 - SAML 2.0 만) "쿼리 매개 변수가 식별자 무시"를 선택합니다.

   이 설정을 사용하면 신뢰 당사자가 AllowCreate 쿼리 매개 변수를 보내 인증 요청에 구성된 AllowCreate 특성 값이 무시됩니다. 식별자 대신 쿼리 매개 변수를 사용하면 파트너 관계 구성을 변경하지 않고도 AllowCreate 특성 값을 변경할 수 있습니다.

   참고: 아이덴티티 공급자가 이 쿼리 매개 변수 설정을 따르게 하려면 "IDP 가 사용자 식별자를 만들도록 허용" 확인란을 선택하십시오.

4. 나열된 각 디렉터리에 대한 디렉터리 검색 사양을 지정합니다. 검색 사양의 두 가지 예:

   LDAP 예제

   uid=%s

   ODBC 예제

   name=%s

5. "다음"을 클릭하여 파트너 관계 구성을 계속합니다.

사용자 식별을 위한 AllowCreate 사용(SAML 2.0)

SAML 2.0 AllowCreate 기능은 SP 에서 "사용자 ID" 구성의 선택적 설정입니다. 인증 요청에 AllowCreate 특성을 포함하면 아이덴티티 공급자가 SP 에 대한 사용자 식별자를 생성할 수 있게 됩니다.

SP 는 아이덴티티 공급자에 인증 요청을 전송하여 싱글 사인온을 시작할 수 있습니다. 요청의 일환으로 서비스 공급자는 true 로 설정된 AllowCreate 특성을 포함할 수 있습니다. 서비스 공급자는 사용자의 아이덴티티를 가져오려고 합니다. AuthnRequest 를 받으면 아이덴티티 공급자가 어설션을 생성합니다. 아이덴티티 공급자가 적절한 사용자 레코드에서 이름 ID 역할을 하는 어설션 특성을 검색합니다. 아이덴티티 공급자가 NameID 특성에 대한 값을 찾을 수 없으면 NameID 에 대한 고유한 영구 식별자를 생성합니다. 아이덴티티 공급자에서 "허용/만들기" 기능을 활성화하여 식별자를 생성하도록 합니다. 아이덴티티 공급자는 고유한 식별자가 있는 어설션을 SP 로 반환합니다.

AllowCreate 쿼리 매개 변수가 사용되도록 설정하여 AllowCreate 특성의 값을 대체할 수 있습니다. 쿼리 매개 변수를 사용하면 파트너 관계를 비활성화하고, 편집하여 다시 활성화하지 않아도 구성된 AllowCreate 설정을 무시할 수 있습니다. 쿼리 매개 변수는 기능을 더욱 유연하게 구현할 수 있게 만듭니다.

제 10 장: 어설션 당사자에서의 어설션 구성

이 섹션은 다음 항목을 포함하고 있습니다.

어설션 구성

파트너 관계 마법사의 "어설션 구성" 단계에서는 다음 설정의 구성을 정의합니다.

이름 ID

필수 어설션 특성인 "이름 ID" 특성은 사용자를 고유한 방식으로 식별합니다. "이름 ID 형식"은 페더레이션된 파트너가 지원하는 식별자 유형을 나타냅니다. "이름 ID 유형"은 이름 ID 형식과 연결된 사용자 프로필 특성을 지정합니다. 사용자 프로필 특성은 사용자 저장소 또는 세션 저장소에서 가져옵니다.

어설션 특성

서블릿, 웹 응용 프로그램 또는 기타 사용자 지정 응용 프로그램에서는 특성을 사용하여 사용자 지정 콘텐츠를 표시하거나 다른 사용자 지정 기능이 사용되도록 설정할 수 있습니다. 특성은 웹 응용 프로그램에서 사용되는 경우 사용자가 신뢰 당사자 측에서 수행하는 작업을 제한할 수 있습니다. 예를 들어 "Authorized Amount"(권한 부여된 금액)이라는 특성 변수는 사용자가 신뢰 당사자 측에서 소비할 수 있는 최대 금액(달러)으로 설정됩니다.

특성은 <AttributeStatement> 요소나 <EncryptedAttribute> 요소에서 지정됩니다. 특성은 이름/값 쌍의 형식을 사용합니다. 특성을 HTTP 헤더나 HTTP 쿠키로 제공할 수도 있습니다.

참고: 특성 명령문은 어설션에 필요하지 않습니다.

특성 명령문 하나에 여러 유형의 특성을 구성할 수 있습니다. 특성 유형은 다음과 같습니다.

- 사용자 특성
- DN 특성
- 정적 데이터
- 세션 특성

  세션 특성 (페이지 99)은 어설션이 세션 저장소에 있는 경우에만 어설션에 사용할 수 있습니다.

어설션 특성을 변환하는 식을 구성할 수도 있습니다. 이 기능을 클레임 변환 (페이지 107)이라고 합니다.

어설션을 받는 경우 신뢰 당사자는 특성 값을 응용 프로그램에 제공합니다.

어설션 생성기 플러그인

일반적으로 특성은 사용자 디렉터리 레코드에서 가져오지만 어설션에는 외부 데이터베이스 또는 응용 프로그램 콘텐츠와 같은 다른 출처의 특성이 포함될 수 있습니다. 다양한 출처에서 특성을 가져오는 어설션 생성기 플러그인을 작성할 수 있습니다. 어설션 생성기 플러그인은 어설션 생성기 플러그인 인터페이스에 따라 작성하는 사용자 지정 코드입니다.

플러그인 작성에 대한 자세한 내용은 *Programming Guide for the Federation Java SDK*(Federation Java SDK 프로그래밍 안내서)를 참조하십시오.

어설션 옵션 구성

어설션 당사자에서 어설션 옵션을 구성하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 "어설션 구성" 단계로 이동합니다.
2. "이름 ID" 섹션에서 설정을 구성합니다.

   신뢰 당사자는 이러한 값을 사용하여 어설션에서 이름 ID 값을 해석합니다.

   선택한 "이름 ID 유형" 옵션에 따라 올바른 값으로 입력을 완료합니다.

   정적 특성

   "값" 필드에 임의 상수 문자열을 입력하십시오.

   사용자 특성

   "값" 필드에 올바른 사용자 저장소 특성을 입력하십시오. 예: 메일

   세션 특성

   "값" 필드에 올바른 세션 저장소 특성을 입력하십시오.

   DN 특성(LDAP 에만 해당)

   "값" 필드에 올바른 LDAP 사용자 디렉터리 특성을 입력하십시오. 또한, DN 사양 필드에 올바른 DN 을 입력하십시오. 예를 들어, DN 특성은 cn=JaneDoe 이고 사양은 ou=Engineering,o=ca.com 입니다.
3. (선택 사항 - SAML 2.0 만) 어설션 당사자가 "이름 ID"의 값을 생성할 수 있도록 "사용자 식별자의 생성 허용"을 선택합니다. 이 기능이 작동하려면 신뢰 당사자의 AuthnRequest 가 AllowCreate 특성을 포함해야 합니다.

   참고: 이 옵션을 선택하는 경우 "이름 ID 형식"의 값이 "영구 식별자"여야 합니다.

4. (선택 사항) "어설션 특성" 테이블에서 "행 추가"를 클릭하여 어설션에 대한 특성을 하나 이상 지정합니다. 원하는 경우 특성을 암호화할 수 있습니다.

   테이블 작성에 대한 도움이 필요하면 몇 가지 어설션 특성 예 (페이지 98)를 확인하십시오. 특성 테이블의 각 열에 대한 자세한 내용을 보려면 "도움말"을 클릭하십시오.

   참고: LDAP 사용자 저장소 특성의 경우 어설션에 다중값 사용자 특성을 추가할 수 있습니다. 다중값 사용자 특성을 지정하는 방법은 "도움말"에 설명되어 있습니다.

5. (선택 사항) CA SiteMinder® Federation Java SDK 를 사용하여 어설션 생성기 플러그인을 작성한 경우 "어설션 생성기 플러그인" 섹션에서 필드에 데이터를 입력합니다.

   플러그인을 작성하려면 *Programming Guide for the Federation Java SDK*(Federation Java SDK 프로그래밍 안내서)를 참조하십시오.

6. "다음"을 클릭하여 파트너 관계 구성을 계속합니다.

어설션 특성 구성 예

다음 그림에서는 어설션 특성 항목의 몇 가지 예를 보여 줍니다. 이 화면은 SAML 2.0 파트너 관계에 해당합니다. SAML 1.1 화면은 이와 비슷하지만 "검색 방법" 및 "형식" 열이 없고, 대신 "네임스페이스" 열이 있습니다.

참고: DN 특성 예에는 항목이 ou=Engineering,o=ca.com 인 "DN 사양" 열이 포함됩니다. 이 열은 이 그림에 표시되어 있지 않습니다.

세션 특성을 어설션에 추가하는 방법

정책 서버는 사용자가 인증된 후 동적 사용자 정보를 유지하기 위해 세션 저장소를 사용합니다. 예를 들면 인증 컨텍스트 정보, SAML 특성, 사용자를 인증하는 타사 IdP, OAuth 인증의 클레임과 같은 정보가 저장됩니다. 정책 서버는 이 정보를 사용하여 사용자 토큰을 생성하거나 정책을 결정할 수 있습니다.

페더레이션된 싱글 사인온의 경우 정책 서버에서 세션 저장소의 특성을 어설션에 추가하여 요청된 응용 프로그램을 사용자 지정할 수 있습니다.

세션 특성은 다음과 같은 배포의 경우에 저장됩니다.

- 위임되지 않은 인증 배포

  로컬 시스템 또는 외부 타사에서 사용자를 인증하지만 시스템에서는 이를 로컬 인증으로 간주합니다. 로컬 인증 배포의 경우 싱글 사인온 구성에서 인증 모드가 로컬로 설정되어야 합니다. 또한 액세스 정책으로 인증 URL 을 보호해야 합니다. 액세스 정책의 인증 체계는 세션 특성을 유지하도록 구성됩니다.

- 위임된 인증 배포

  외부 타사에서 사용자를 인증할 수 있습니다. 타사 파트너는 세션 저장소에 저장되는 사용자 정보를 반환합니다.

다음 그림에서는 세션 특성을 구성한 후 어설션에 추가하는 데 필요한 단계를 보여 줍니다.

세션 특성을 지원하려면 다음 단계를 완료하십시오.

1. 사용할 수 있는 세션 특성을 확인합니다. (페이지 101)
2. 세션 특성을 어설션 구성에 추가합니다. (페이지 101)
3. SSO 에 대한 인증 모드 및 URL 을 확인합니다 (페이지 102).
4. 세션 특성을 유지하도록 인증 체계를 구성합니다. (페이지 103)
5. 인증 URL 을 보호하는 정책을 생성합니다. (페이지 104)

사용할 수 있는 세션 특성 확인

페더레이션 관리자는 파트너 관계에 사용되는 세션 특성을 식별해야 합니다. 사용 가능한 특성을 쉽게 파악할 수 있도록 데이터베이스나 사용자 디렉터리 같은 인증 원본을 사용하십시오.

세션 특성을 어설션 구성에 추가

세션 특성을 어설션 구성에 추가합니다. 구성은 IdP-SP 파트너 관계 같은 어설션 당사자 측에서 수행됩니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. 파트너 관계 마법사의 "어설션 구성" 단계로 이동합니다.
3. "어설션 특성" 섹션에서 "행 추가"를 클릭합니다.
4. 세션 특성을 구성하려면 테이블 설정을 완료합니다. 예를 들면 다음과 같습니다.

   어설션 특성

   IssuerID

   검색 방법

   SSO

   형식

   지정되지 않음

   유형

   세션 특성

   값

   IssuerID

   특성 테이블에 대한 자세한 내용을 보려면 "도움말"을 클릭하십시오.
5. 필요한 항목 수에 맞춰 행을 추가합니다.
6. (선택 사항) 특성을 암호화하려면 "암호화"를 선택합니다.
7. "다음"을 클릭하여 "SSO 및 SLO" 단계로 이동합니다.

관리 UI 의 세션 특성 예

다음 그림의 마지막 두 개 항목은 세션 특성 항목의 예를 보여 줍니다. 이 화면은 SAML 2.0 파트너 관계에 해당합니다. SAML 1.1 화면은 이와 비슷하지만 "검색 방법" 및 "형식" 열이 없고, 대신 "네임스페이스" 열이 있습니다.


어설션 특성

어설션 특성

어설션 특성	검색 방법	형식	유형	값
region	SSO	Unspecified	Static	northeast
email	SSO	Unspecified	User Attribute	mail
admintitle	SSO	Unspecified	Expression	== 'Manager' ? 'Administrato
dn	SSO	Unspecified	DN Attribute	cn=JaneDoe
IssuerDN	SSO	Unspecified	Session Attribute	Issuer DN
SubjectDN	SSO	Unspecified	Session Attribute	Subject DN

SSO 에 대한 인증 모드 및 URL 확인

파트너 관계의 인증 모드 및 인증 URL 이 올바르게 설정되어 있는지 확인하십시오.

참고: 이 절차에서는 필요한 다른 SSO 설정이 구성되어 있다고 가정합니다.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
2. "인증" 섹션에서 다음 필드의 설정을 확인합니다.

   인증 모드

   로컬

인증 URL

이 URL 은 redirect.jsp

파일(예: http://*myserver*.idpA.com/siteminderagent/redirectjsp/redirect.jsp)을 가리킵니다.

myserver

웹 에이전트 옵션 팩 또는 SPS 페더레이션 게이트웨이가 포함된 웹 서버를 식별합니다. redirect.jsp 파일은 어설션 당사자에 설치된 웹 에이전트 옵션 팩 또는 SPS 페더레이션 게이트웨이에 포함되어 있습니다.

정책으로 이 리소스를 보호합니다.

3. "확인" 단계로 이동하고 "마침"을 클릭합니다.

세션 특성을 유지하도록 인증 체계 구성

인증 URL 을 보호하는 인증 체계를 구성하고 체계에서 세션 특성을 유지하게 합니다. 이 절차는 시스템이 세션 특성을 저장하도록 하는 데 필요합니다.

다음 단계를 수행하십시오.

1. "인프라", "인증", "인증 체계"를 차례로 클릭합니다.
2. "인증 체계 만들기"를 클릭합니다.
3. "인증 체계 유형의 새 개체 만들기"가 선택되어 있는지 확인합니다. "확인"을 클릭합니다.

   "인증 체계 만들기" 페이지가 표시됩니다.

4. 세션 특성을 유지할 수 있는 인증 체계 템플릿을 선택합니다. 이러한 인증 체계 템플릿에는 사용자 이름 및 암호 외에도 추가 정보가 필요합니다.

   예를 들어 X.509 인증서 인증 체계에는 인증서에 대한 SubjectDN 및 IssuerID 가 필요합니다. OAuth 인증 체계에는 이름 및 성과 같은 정보가 필요합니다. 이 정보는 세션 저장소에서 유지되었다가 어설션에 추가될 수 있습니다.

사용할 수 있는 인증 체계 템플릿은 다음과 같습니다.

- OpenID
- OAuth
- 모든 X.509 인증 템플릿
- 사용자 지정 체계

5. 체계와 관련된 필드 및 컨트롤을 완료합니다.

   필드 설명을 보려면 "도움말"을 클릭하십시오.

6. 대화 상자의 "체계 설정" 섹션에서 "인증 세션 변수 유지"를 선택합니다.
7. "제출"을 클릭하여 체계를 저장합니다.

인증 URL 을 보호하는 정책 생성

인증 URL 을 보호하는 정책에서 세션 특성을 유지하는 인증 체계를 사용하십시오. 사용자가 보호된 리소스를 요청하면 정책에 의해 사용자를 인증하는 데 필요한 작업이 트리거됩니다. 사용자가 세션 변수로 제공하는 자격 증명이 저장됩니다.

먼저 어설션 당사자와 할당 사용자에 대한 정책 도메인을 생성하십시오. 기존 어설션 당사자 도메인을 수정할 수도 있습니다.

다음 단계를 수행하십시오.

1. "정책", "도메인", "도메인"을 클릭합니다.

   "도메인" 페이지가 표시됩니다.

2. 해당 어설션 당사자의 도메인을 선택하고 수정합니다.
3. 사용자 디렉터리가 도메인의 일부인지 확인합니다. 그렇지 않으면 "추가/제거"를 클릭하여 사용자 디렉터리를 추가합니다.

   "사용 가능한 구성원" 목록에서 사용자 디렉터리를 하나 이상 선택할 수 있습니다. 한 번에 여러 구성원을 선택하려면 Ctrl 키를 누른 채 추가 구성원을 클릭하십시오. 구성원 블록을 선택하려면 첫 번째 구성원을 클릭한 다음 Shift 키를 누른 채 블록의 마지막 구성원을 클릭하십시오.

   참고: 사용자 디렉터리를 생성하여 도메인에 추가하려면 "만들기"를 클릭하십시오.

4. "제출"을 클릭합니다.

도메인이 구성되었습니다.

인증 URL 정책에 대한 영역 및 규칙 생성

페더레이션 도메인의 경우 영역을 생성하여 웹 에이전트와 연결하십시오.

다음 단계를 수행하십시오.

1. "정책", "도메인", "영역"을 클릭합니다.

   "영역" 페이지가 표시됩니다.

2. "영역 만들기"를 클릭합니다.
3. 수정할 도메인을 선택하고 "다음"을 클릭합니다.
4. 영역의 이름과 설명을 입력합니다.

   해당 영역이 SSO 인증 URL 을 위한 영역임을 나타내는 이름을 지정합니다.

5. "에이전트/에이전트 그룹 조회"를 클릭하여 에이전트를 선택합니다.
6. 적절한 웹 에이전트를 선택하고 "확인"을 클릭합니다.
7. redirect.jsp 에 대한 리소스 필터를 지정합니다. 예를 들면 다음과 같습니다.

   /siteminder/redirectjsp/redirect.jsp

8. 나머지 필드를 완료합니다.

   기본 리소스 보호

   보호됨

   인증 체계

   인증 URL 을 보호하기 위해 구성한 인증 체계를 선택합니다. 이 체계는 세션 특성을 유지하기 위해 구성한 체계입니다.

9. "규칙" 섹션에서 규칙을 생성합니다.
   a. 규칙의 이름을 지정합니다.
   b. 나머지 설정은 기본값을 사용합니다.
10. 다른 구성 옵션은 건너뜁니다.
11. "마침"을 클릭합니다.

영역 및 규칙 구성이 완료되었습니다.

인증 URL 정책 완성

인증 URL 을 보호하는 정책을 생성하십시오 정책 구성 요소는 함께 작동하여 리소스를 보호합니다.

정책을 생성한 후에는 사용자와 규칙을 추가하십시오.

다음 단계를 수행하십시오.

1. "정책", "도메인", "도메인"을 클릭합니다.
2. 도메인을 검색합니다.

   검색 조건과 일치하는 도메인의 목록이 표시됩니다.
3. 어설션 당사자의 도메인을 선택합니다.
4. "수정"을 클릭합니다.
5. "정책" 탭을 클릭합니다.

   "정책" 페이지가 표시됩니다.
6. "만들기"를 클릭합니다.
7. 정책의 이름과 설명을 입력합니다.
8. "사용자" 탭에서 개별 사용자, 사용자 그룹 또는 둘 모두를 추가합니다. 사용자는 도메인과 연결된 사용자 디렉터리의 구성원입니다.

   각 사용자 디렉터리 그룹 상자 내에서 "구성원 추가", "항목 추가" 및 "모두 추가"를 선택합니다. 사용하는 방법에 따라 사용자를 추가할 수 있는 대화 상자가 열립니다.

   참고: "구성원 추가"를 선택하는 경우 "사용자/그룹" 창이 열립니다. 개별 사용자는 자동으로 표시되지 않습니다. 한 디렉터리 내의 특정 사용자를 찾으려면 검색 유틸리티를 사용하십시오.

   오른쪽 화살표(>)를 클릭하여 사용자 또는 그룹을 편집하거나 빼기 기호(-)를 클릭하여 사용자 또는 그룹을 삭제할 수 있습니다.
9. "규칙" 탭에서 규칙을 추가합니다.

10. 인증 URL 에 대해 생성한 규칙을 선택하고 "확인"을 클릭합니다.

    규칙에 대한 응답은 구성할 필요가 없습니다.

11. "제출"을 클릭하여 구성을 완료합니다.

정책 구성이 완료되었습니다.

어설션 특성, 싱글 사인온 및 정책 구성이 함께 작동하여 세션 특성을 어설션에 사용할 수 있게 만듭니다.

어설션 당사자에서 클레임 변환을 구성하는 방법

클레임 변환 기능은 페더레이션된 싱글 사인온 트랜잭션 동안 클레임을 조작합니다. 클레임은 특성이라고도 하며, 특성을 사용자 지정하고 파트너의 사용자 환경을 개선하는 데 유용합니다.

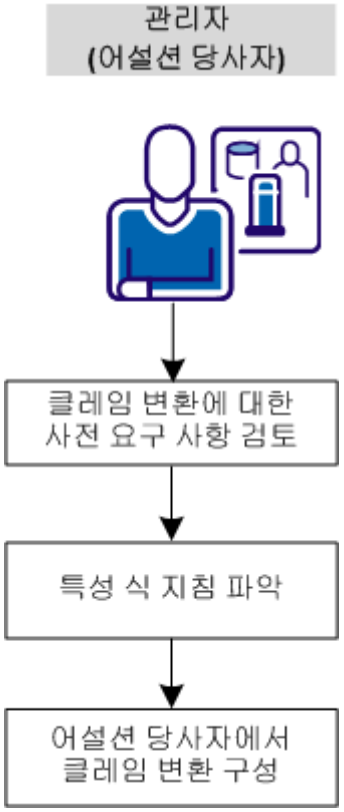
어설션 특성을 수정하면 신뢰 당사자는 대상 응용 프로그램에서 사용할 수 있도록 사용자 정보를 조정할 수 있게 됩니다. 예를 들어 클레임 변환을 통해 서로 다른 도메인에 있는 서로 다른 파트너에서의 역할을 연결할 수 있습니다. 한 도메인에서 사용자가 엔지니어링 관리자이고 EngineerAdmins 라는 그룹에 속하지만, 신뢰 당사자는 동일한 역할을 DevelAdmins 로 식별할 수 있습니다. 이 경우 어설션 당사자는 어설션을 발급하기 전에 역할 특성을 변경합니다. 그러면 해당 사용자가 신뢰 당사자 응용 프로그램에서 인식될 수 있는 DevelAdmins 역할로 식별됩니다.

클레임 변환은 로컬 어설션 당사자에서 어설션 생성 프로세스 도중에 발생합니다. 이 기능은 파트너 관계별로 구성해야 합니다. 어설션을 로컬 당사자가 생성하든 원격 당사자가 생성하든 관계없이 어설션을 수정할 수 있습니다. 클레임은 파트너 관계에 대해 구성된 식을 기반으로 변환됩니다. 식에서는 사용자 저장소와 SiteMinder 세션 저장소에서 가져온 사용자 정보를 사용합니다.

소프트웨어에서는 어설션 특성에 대해 다음과 같은 세 가지 수정 작업을 수행할 수 있습니다.

- **변환:** 어설션 특성의 값을 다른 값으로 변경합니다.
- **추가:** 어설션 특성이 아직 없는 경우 추가합니다.
- **삭제:** 조건에 따라 어설션 특성을 삭제합니다.

다음 그림에서는 구성 단계를 보여 줍니다.



클레임 변환을 설정하려면 다음 단계를 수행하십시오.

1. 클레임 변환을 위한 사전 요구 사항을 검토합니다. (페이지 108)
2. 특성 식 지침을 확인합니다. (페이지 109)
3. 어설션 당사자에서 클레임 변환을 구성합니다. (페이지 110)

클레임 변환에 대한 사전 요구 사항

클레임 변환을 구성하기 전에 다음 사전 요구 사항을 고려하십시오.

- 사용할 수 있는 사용자 저장소 및 세션 저장소 특성을 잘 알고 있어야 합니다.
- 신뢰 당사자가 어설션에서 검색할 것으로 예상되는 특성을 확인해야 합니다.
- UEL(Unified Expression Language)의 오픈 소스 버전인 JUEL(Java Unified Expression Language)을 잘 알고 있어야 합니다.

특성 식 지침 확인

식은 소프트웨어에 어설션 특성의 조작 방법을 알려 주는 규칙입니다. 식이 전달하는 지침을 통해 소프트웨어에서 어설션 특성을 수정, 추가 또는 삭제하게 됩니다. JUEL(Java Unified Expression Language)을 사용하여 식을 구성하십시오. JUEL 식 계산기는 구성된 식을 검사하고 결과 어설션 특성을 생성합니다.

관리 UI 의 "어설션 특성" 테이블에서 식을 정의하십시오. 이 테이블에 액세스하려면 파트너 관계 마법사의 "어설션 구성" 단계로 이동하십시오. 이 테이블은 다음 그림에 표시되어 있습니다.

어설션 특성 테이블의 "Value"(값) 열에 식을 입력하십시오. 식의 모든 특성은 사용자 저장소 또는 세션 저장소 특성입니다.

일반적으로 식은 조건에 따라 작동합니다. 조건이 충족되면 지정된 클레임 수정이 수행됩니다. 예를 들어 수신 어설션에 "role" 특성이 포함된 경우 "role" 어설션 특성을 수정하는 식은 다음과 같습니다.

#{attr["title"] == 'manager' ? 'administrator' : attr["title"]}

표현식의 첫 번째 부분 **#{attr["title"] == 'manager'**는 소프트웨어로 하여금 로그인한 사용자의 직책이 "manager"인지 확인하도록 합니다. 조회는 사용자 디렉터리에서 수행됩니다. 이 조건이 충족되면 표현식의 두 번째 부분 **? 'administrator' :**가 role 어설션 특성에 값 "administrator"를 할당합니다. 조건이 충족되지 않을 경우 식의 마지막 부분 **attr["title"]}**은 사용자 특성 "title"의 값이 "manager"로 유지되도록 합니다. 어설션 특성 "role"에는 이 "manager" 값이 할당됩니다.

참고: 식에서 attr["title"] 구문 대신 앞의 예에 나온 'administrator'와 같이 정적 값을 사용할 수 있습니다.

이 예에서는 어설션에 "role" 특성이 이미 있다고 가정합니다. 따라서 이 식은 기존 특성을 변환하는 식입니다. "role"이 어설션에 포함되지 않은 경우 소프트웨어는 role 특성을 어설션에 추가합니다.

식 구문

다음과 같은 올바른 구문을 사용하여 식을 구성하십시오.

- 사용자 저장소 특성은 **attr**["*attribute_name*"] 문자열로 나타냅니다.
- 세션 저장소 특성은 **session_attr**["*attribute_name*"] 문자열로 나타냅니다.
- 클레임을 삭제하려면 'DELETE' 인수를 사용합니다.

attr 및 **session_attr** 접두사에는 소문자 텍스트를 사용합니다. 특성 이름은 대/소문자를 구분하지 않습니다.

또한 다음과 같은 조건부 JUEL 연산자에 대해 잘 알고 있어야 합니다.

연산자	의미
conditional value ? value1 : value2	conditional value 가 value1 또는 value2 가 됩니다.
!=	같지 않음
==	같음

중요! 식의 특성은 사용자 디렉터리나 세션 저장소에서 사용할 수 있는 특성이어야 합니다. 특성이 올바르지 않으면 시스템에서는 단순히 해당 특성에 빈 값을 포함합니다. 어설션 생성은 오류 없이 실행됩니다.

더 많은 식의 예를 보려면 어설션 당사자 측에서 클레임 변환 구성 (페이지 110) 단원을 읽어 보십시오.

어설션 당사자 측에서 클레임 변환 구성

파트너 관계 수준에서 식을 정의하십시오. 이러한 식의 결과로 어설션에서 특성이 수정, 추가 또는 삭제됩니다. 규칙이 정의된 후에는 어설션이 수정되어 신뢰 당사자로 보내집니다. 클레임 변환을 구성하지 않으면 어설션 특성이 신뢰 당사자에게 "그대로" 전달됩니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "페더레이션", "파트너 관계"를 차례로 선택합니다.

3. 수정할 파트너 관계를 선택합니다. 적합한 파트너 관계로는 다음이 포함됩니다.
   - 로컬 생산자-원격 소비자
   - 로컬 IdP-원격 SP
   - 로컬 IP-원격 RP
4. 파트너 관계 마법사의 "어설션 구성" 단계로 이동합니다.

   "어설션 특성" 섹션에서 "행 추가"를 클릭합니다.
5. 추가한 행에서 다음 필드에 특히 주의해야 합니다. 각 필드에 대한 자세한 설명을 보려면 "도움말"을 클릭합니다.

   어설션 특성

   어설션 특성을 입력합니다. 이 열의 모든 값은 어설션 특성입니다. 어설션의 기존 특성은 어설션에 유지되지만 해당 값은 구성된 식에 따라 새로 설정됩니다. DELETE 식을 구성한 경우에만 어설션에서 특성이 제거됩니다.

   검색 방법

   기본값 SSO 를 유지합니다.

   형식

   어설션에 추가되는 특성의 형식을 지정합니다. 형식 옵션은 엔터티에 대한 SAML 프로필에 따라 다릅니다.

   유형

   식

   클레임 변환에는 항상 이 값을 사용합니다.

   값

   어설션 특성을 수정할 방법을 반영하는 식을 입력합니다.

   클레임 식 구성 (페이지 109)에 대한 지침과 다음 예를 검토하십시오.
   - 어설션의 클레임 변환 (페이지 112)
   - 어설션에 클레임 추가 (페이지 114)
   - 어설션에서 클레임 삭제 (페이지 116)

6. (SAML 2.0 및 토큰 유형이 SAML 2.0 인 WSFED 의 경우 선택 사항) 어설션 특성을 암호화하려면 "암호화"를 선택합니다. 어설션 당사자는 파트너 관계 구성에 지정된 인증서를 사용하여 어설션을 암호화합니다.

   신뢰 당사자는 인증서와 연결된 개인 키를 사용하여 어설션 특성의 암호를 해독합니다.

7. 구성하려는 어설션 특성에 필요한 만큼 행을 추가합니다.

클레임 변환이 파트너 관계에 구성된 항목을 기준으로 구현됩니다.

어설션의 클레임 변환

클레임을 변환하면 어설션 특성 값이 다른 값으로 변경됩니다.

참고: 아래 예에서는 "어설션 특성", "유형" 및 "값"에 대한 항목만 보여 줍니다.

변환 예 1

다음 예에서는 어설션에 "title" 특성이 이미 있다고 가정합니다. 표에는 사용자 저장소의 사용자 특성이 나와 있습니다.

사용자 디렉터리 특성	특성 값
role	admin
admintitle	SeniorAdmin
supertitle	SuperUser

다음 구성을 사용하여 기존 title 특성의 값을 변환할 수 있습니다.

어설션 특성

title

유형

식

값

#{attr["role"] == 'admin' ? attr["admintitle"] : attr["supertitle"]}

결과: 이 식은 "role" 사용자 특성이 "admin"으로 설정되어 있는지 여부를 조건으로 합니다. 이 조건이 충족될 경우 어설션 특성 "title"이 "admintitle" 특성 값 SeniorAdmin 으로 설정됩니다. role 특성이 "admin"이 아닌 다른 값으로 설정되어 있으면 "title" 특성은 "supertitle" 특성 값인 SuperUser 가 됩니다.

변환 예 2

다음 예에서는 어설션에 ContactNo 특성이 이미 있다고 가정합니다.

사용자 디렉터리 특성	특성 값
homephone	555-3344
mobile	555-8888

다음 구성을 사용하여 기존 title 특성의 값을 변환할 수 있습니다.

어설션 특성

ContactNo

유형

식

값

#{attr["homephone"] == '555-3344' ? attr["mobile"] : attr["homephone"]}

결과: 이 식은 로그인한 사용자의 "homephone" 사용자 특성이 555-3344 로 설정되어 있는지 여부를 조건으로 합니다. 이 조건이 충족될 경우 어설션 특성은 "mobile" 특성 값인 555-8888 로 설정됩니다. 조건이 충족되지 않으면 "homephone" 값이 변경되지 않습니다.

참고: 세션 특성을 사용하는 식을 구성하려면 attr["*attribute_name*"]을 session_attr["*attribute_name*"]로 바꾸십시오. 예:

#{session_attr["att1"] == 'admin' ? session_attr["attr2"] : attr["attr3"]}

어설션에 클레임 추가

아직 없는 어설션 특성을 추가할 수 있습니다.

추가 예 1

다음 예에서는 어설션에 "title" 특성이 *없다고* 가정합니다.

사용자 디렉터리 특성	특성 값
role	admin
admintitle	director
supertitle	executive

다음 구성으로 어설션에 title 특성을 추가할 수 있습니다.

어설션 특성

title

유형

식

값

#{attr["role"] == 'admin' ? attr["admintitle"] : attr["supertitle"]}

결과: 이 식은 로그인한 사용자의 "role" 특성이 admin 으로 설정되어 있는지 여부를 조건으로 합니다. 이 조건이 충족될 경우 어설션 특성 "title"이 어설션에 추가되고 "admintitle" 특성 값인 "director" 값으로 설정됩니다. role 특성이 "admin"이 아닌 다른 값으로 설정되어 있으면 어설션 특성 "title"이 추가되지만 해당 값은 "supertitle" 특성 값인 "executive"가 됩니다.

추가 예 2

다음 예에서는 어설션에 "smtitle" 특성이 *없다고* 가정합니다.

사용자 디렉터리 특성	특성 값
title	관리자

어설션 특성

smtitle

유형

식

값

#{attr["title"] == 'manager' ? 'federation administrator' : attr["title"]}

결과: 로그인한 사용자의 title 특성이 "manager"인 경우 "smtitle"이 어설션에 추가되고 해당 값은 "federation administrator"로 설정됩니다. 물음표 뒤에는 attr["attribute_name"] 구문을 사용하는 대신 정적 값을 입력할 수 있습니다. 이 예에서는 federation administrator 가 정적 값에 해당합니다.

참고: 세션 특성을 사용하는 식을 구성하려면 attr["*attribute_name*"]을 session_attr["*attribute_name*"]로 바꾸십시오. 예:

#{session_attr["att1"] == 'admin' ? session_attr["attr2"] : attr["attr3"]}

어설션에서 클레임 삭제

어설션 특성을 삭제할 수 있습니다.

삭제 예 1

두 개의 항목을 구성하여 어설션 특성 admintitle 및 supertitle 을 삭제할 수 있습니다.

사용자 디렉터리 특성	특성 값
role	admin 또는 superuser
title	administrator
su	superuser

어설션 특성

admintitle

유형

식

값

#{attr["role"] == 'superuser' ? 'DELETE' : attr["title"]}

결과: 이 식 문자열은 "role" 사용자 특성을 조건으로 합니다. 로그인한 사용자의 role 특성이 superuser 면 어설션 특성 "admintitle"이 삭제되고, role 특성이 superuser 가 아니면 title 어설션 특성 값이 title 사용자 디렉터리 특성의 값인 administrator 로 설정됩니다.

어설션 특성

supertitle

유형

식

값

#{attr[“role”] == ‘admin’ ? 'DELETE' : attr["su"]}

결과: 이 식 문자열은 "role" 사용자 특성을 조건으로 합니다. 로그인한 사용자의 role 특성이 "admin"이면 어설션 특성 "supertitle"이 삭제되고, role 특성이 "admin"이 아니면 supertitle 어설션 특성 값이 su 사용자 디렉터리 특성의 값인 superuser 로 설정됩니다.

삭제 예 2

다음 예에서는 식 하나로 추가와 삭제를 결합하는 경우를 보여 줍니다.

사용자 디렉터리 특성	특성 값
title	관리자

어설션 특성

ManagerName

유형

식

값

#{attr["title"] != 'Manager' ? attr["manager"] : 'DELETE'}

결과: 로그인한 사용자의 사용자 특성 title 이 "manager"가 *아니면* ManagerName 특성이 어설션에 추가됩니다. 하지만 로그인한 사용자의 title 이 manager 이면 ManagerName 특성은 어설션에 포함된 것으로 가정하여 삭제됩니다.

참고: 세션 특성을 사용하는 식을 구성하려면 attr["*attribute_name*"]을 session_attr["*attribute_name*"]로 바꾸십시오. 예:

#{session_attr["att1"] == 'admin' ? session_attr["attr2"] : attr["attr3"]}

어설션 콘텐츠 사용자 지정

AssertionGeneratorPlugin 인터페이스 구현

사용자 지정 어설션 생성기 플러그인을 생성할 때의 첫 번째 단계는 AssertionGeneratorPlugin 인터페이스를 구현하는 것입니다. 다음 요구 사항이 구현 클래스에 적용됩니다.

- 구현에서는 매개 변수가 포함되지 않은 공개 기본 생성자 메서드를 제공해야 합니다.
- 구현은 상태 비저장이어야 합니다. 따라서 여러 스레드가 단일 플러그인 클래스를 사용할 수 있어야 합니다.
- 구현에 customizeAssertion 메서드 호출이 포함되어야 합니다. 요구 사항에 따라 이러한 메서드의 기존 구현을 덮어쓸 수 있습니다. 샘플 프로그램을 참조하십시오.
- customizeAssertion 메서드에 전달되는 매개 변수 문자열의 구문 요구 사항과 사용에 대한 책임은 사용자 지정 개체에 있습니다.

참고: *federation_sdk_home*\sample\com\ca\federation\sdk\plugin\sample 폴더에 두 개의 샘플 구현 클래스가 포함되어 있습니다.

어설션 생성기 플러그인 배포

AssertionGeneratoPlugin 인터페이스에 대한 구현 클래스를 코드로 지정한 다음에는 해당 구현 클래스를 컴파일하고 CA SiteMinder?Federation 이 실행 파일을 찾을 수 있는지 확인하십시오.

어설션 생성기 플러그인을 배포하려면

1. 다음 방법 중 하나로 어설션 플러그인 코드를 컴파일합니다.
   - 샘플 플러그인을 사용하는 경우 플랫폼의 빌드 스크립트를 사용하여 플러그인을 컴파일합니다. 빌드 스크립트는 *federation_sdk_home*\sample 디렉터리에 설치됩니다. 빌드 스크립트는 다음과 같습니다.

     Windows: build_plugin.bat

     UNIX: build_plugin.sh

     컴파일된 샘플 플러그인 fedpluginsample.jar 은 *federation_sdk_home*\jar 디렉터리에 있습니다.
   - 플러그인을 직접 작성하는 경우에는 플러그인을 컴파일할 때 smapi.jar 를 포함하십시오.
2. JVMOptions.txt 파일에서 플러그인의 클래스 경로를 포함하도록 -Djava.class.path 값을 수정합니다.
   federation_install_dir\siteminder\config 디렉터리에서 JVMOptions.txt 파일을 찾습니다.

   플러그인 jar 를 원하는 디렉터리에 저장하고 JVMOptions.txt 파일에서 해당 위치를 가리키도록 할 수 있습니다. 샘플 플러그인을 사용하려면 fedpluginsample.jar 를 가리키도록 클래스 경로를 수정해야 하지만 smapi.jar 의 클래스 경로는 수정하지 마십시오.

   참고: 플러그인에서 Apache Xerces 또는 Xalan 을 사용하려면 CA SiteMinder® Federation 과 함께 설치된 Xerces 또는 Xalan 바이너리 파일을 사용하십시오. 이 바이너리 파일은 CA SiteMinder® Federation SDK 와 함께 설치되지 않습니다. 이 파일은 호환성을 위해 필요합니다.
3. CA SiteMinder® Federation 서비스를 다시 시작합니다.

   서비스를 다시 시작하면 CA SiteMinder® Federation 이 최신 버전의 어설션 생성기 플러그인을 사용합니다.

어설션 생성기 플러그인이 사용되도록 설정

어설션 생성기 플러그인을 작성하고 컴파일한 후 CA SiteMinder?Federation UI 에서 설정을 구성하여 플러그인이 사용되도록 설정해야 합니다. UI 매개변수는 CA SiteMinder?Federation 에게 플러그인을 찾을 수 있는 위치를 알려줍니다.

플러그인을 배포 (페이지 118)할 때까지 플러그인 설정을 구성하지 마십시오.

어설션 생성기 플러그인이 사용되도록 설정하려면

1. 관리 UI 에 로그온합니다.
2. 수정하려는 파트너 관계에 대한 파트너 관계 마법사의 "어설션 구성" 단계로 이동합니다.
3. 다음 어설션 생성기 플러그인 설정의 값을 입력합니다.

플러그인 클래스

플러그인의 Java 클래스 이름을 지정합니다. 이름을 입력합니다. 이 플러그인은 런타임에 호출됩니다.

예: com.mycompany.assertiongenerator.AssertionSample

플러그인 클래스는 어설션을 구문 분석 및 수정한 다음 최종 처리를 위해 결과를 CA SiteMinder® Federation 으로 반환할 수 있습니다. 각 신뢰 당사자의 어설션 생성기 플러그인을 지정합니다. SDK 에 컴파일된 샘플 플러그인이 포함되어 있습니다. 컴파일된 샘플 어설션 플러그인은 *federation_sdk_home*/jar 디렉터리에서 볼 수 있습니다.

참고:
federation_sdk_home\sample\com\ca\federation\sdk\plugin\sample 디렉터리에서 CA SiteMinder® Federation 샘플 플러그인의 소스 코드도 볼 수 있습니다.

플러그인 매개 변수

(선택 사항) CA SiteMinder® Federation 이 런타임에 플러그인에 매개변수로 전달하는 문자열을 지정합니다. 이 문자열에는 어떤 값이든 포함될 수 있으며 따라야 하는 특정 구문이 없습니다.

플러그인은 수신하는 매개 변수를 해석합니다. 예를 들어 매개 변수는 특성 이름일 수도 있고 플러그인에 작업을 수행하도록 지시하는 정수가 문자열에 포함될 수도 있습니다.

참조 정보(메서드 서명, 매개 변수, 반환 값, 데이터 형식)와 UserContext 클래스 및 APIContext 클래스에 대한 생성자는 *Javadoc 참조서*에서 확인할 수 있습니다. Javadoc 의 AssertionGeneratorPlugin 인터페이스를 참조하십시오.

제 11 장: 싱글 사인온 구성

이 섹션은 다음 항목을 포함하고 있습니다.

싱글 사인온 구성(어설션 당사자)

신뢰 당사자에 어설션이 제공되는 방식을 지정하려면 어설션 당사자 측에서 싱글 사인온을 구성하십시오.

다음 절차에서는 싱글 사인온이 사용되도록 설정하기 위한 기본 단계를 설명합니다. 사인온 대화 상자의 구성 가능한 모든 기능에 대한 자세한 내용은 이후 항목과 관리 UI 도움말에 설명되어 있습니다.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 적절한 단계에서 시작합니다.

   SAML 1.1

   싱글 사인온

   SAML 2.0

   SSO 및 SLO

   WSFED

   싱글 사인온 및 사인아웃

   원격 신뢰 당사자를 생성하거나 가져올 때 정의된 모든 값이 입력됩니다.

2. 다음 정보를 참고하여 "인증" 섹션의 필드를 채우십시오.

   - "인증 모드" 필드에서 "로컬"을 선택할 경우 "인증 URL"에 redirect.jsp 파일을 가리키는 URL 을 입력하십시오. 예를 들면 다음과 같습니다.

     http://webserver1.example.com/affwebservices/redirectjsp/redirect.jsp

     이 예에서 webserver1 은 웹 에이전트 옵션 팩이 있는 웹 서버를 식별합니다. redirect.jsp 파일은 아이덴티티 공급자 사이트에 설치된 웹 에이전트 옵션 팩에 포함되어 있습니다.

     중요! 액세스 제어 정책을 사용하여 인증 URL 을 보호 (페이지 65)하십시오. 영역, 규칙, 정책을 구성하십시오. 어설션에 세션 정보를 추가하려면 "인증 세션 변수 유지" 확인란을 선택하십시오.

   - 인증 모드로 "위임됨"을 선택할 경우에는 추가 필드를 구성하십시오. 자세한 내용은 위임된 인증 (페이지 207)을 참조하십시오.

3. "인증 클래스" 필드를 채웁니다(SAML 1.1 및 2.0 만). 이 필드에 정적 URI 를 제공합니다. 또한 SAML 2.0 의 경우에 한해 소프트웨어가 인증 클래스를 자동으로 검색할 수 있습니다. URI 는 어설션의 AuthnContextClassRef 요소에 배치되어 사용자 인증 방법을 설명합니다.

4. "SSO" 섹션의 필드에 데이터를 입력합니다. 이 설정을 통해 다음 기능을 제어할 수 있습니다.
   - 싱글 사인온 바인딩
   - 어설션 유효 기간

     "SSO 유효 기간" 및 "차이 시간"에 따라 어설션이 유효한 시기가 결정됩니다. 이러한 설정의 상호 작용 방식을 이해하려면 어설션 유효성 (페이지 131)에 대한 정보를 읽어 보십시오.

   SAML 2.0 의 경우 다음 기능을 구성할 수 있습니다.
   - 싱글 사인온을 시작할 파트너
   - SP 세션 유효 기간
   - SP 세션 기간
   - 사용자 아이덴티티 정보를 SP 와 공유하기 위한 사용자 동의

   필드 설명을 보려면 "도움말"을 클릭하십시오.
5. 어설션 소비자 서비스 또는 보안 토큰 서비스의 URL 을 지정합니다. 이 원격 신뢰 당사자 서비스는 어설션을 소비하고 처리합니다.

   이 URL 은 파트너가 사용자에게 제공해야 합니다.
6. SAML 바인딩으로 HTTP-아티팩트를 선택한 경우 백 채널 설정 (페이지 133)을 구성합니다.
7. (선택 사항) SAML 2.0 의 경우 다음 태스크를 수행할 수 있습니다.
   - IDP 검색 프로필 (페이지 148)이 사용되도록 설정합니다.
   - 지정된 HTTP 오류에 대한 상태 리디렉션 URL (페이지 130)을 지정합니다.

추가 정보:

싱글 사인온을 시작할 수 있는 SAML 2.0 엔터티 (페이지 130)
HTTP 오류에 대한 상태 리디렉션(SAML 2.0 IdP) (페이지 130)
HTTP-아티팩트 백 채널의 레거시 아티팩트 보호 유형 (페이지 126)

파트너 관계 페더레이션의 인증 모드

파트너 관계 페더레이션에서는 페더레이션된 싱글 사인온의 인증 모드를 정의할 수 있습니다.

- **로컬 인증 모드**

  로컬 인증은 주로 로컬 페더레이션 시스템에서 발생합니다. 로컬 인증의 경우 인증 체계로 기본 인증이나 양식 인증을 선택할 수 있습니다. 이 두 옵션은 로컬로 사용할 수 있는 유일한 방법입니다.

  외부 타사에서 사용자를 인증하는 경우 인증 모드로 로컬 인증을 선택할 수도 있습니다. 타사에서 사용자 정보를 전달하면 해당 사용자 정보는 나중에 어설션에 사용할 수 있도록 세션 저장소에 저장됩니다.

- **위임된 인증 모드**

  위임된 인증에서는 타사 WAM(웹 액세스 관리) 시스템에 인증 태스크를 전달합니다. 타사에서 사용자를 인증하는 데 사용되는 방법은 해당 타사가 지원하는 인증 체계에 따라 달라집니다. 타사 WAM 은 사용자를 인증한 후 페더레이션된 사용자 아이덴티티를 SiteMinder 로 다시 보냅니다.

HTTP-아티팩트 백 채널의 레거시 아티팩트 보호 유형

HTTP-아티팩트 싱글 사인온의 경우 "아티팩트 보호 유형" 필드에 대해 레거시 옵션을 선택할 수 있습니다. 레거시 옵션은 어설션 당사자의 아티팩트 서비스에 대한 백 채널을 보호하는 레거시 방법을 사용 중임을 나타냅니다.

레거시 보호 방법을 구현하려면

- 에이전트 그룹 FederationWebServicesAgentGroup 에 FWS 응용 프로그램을 보호하는 웹 에이전트를 추가하십시오.
  - ServletExec 의 경우 이 에이전트는 웹 에이전트 옵션 팩이 설치된 웹 서버에 있습니다.
  - WebLogic 이나 JBOSS 와 같은 응용 프로그램 서버의 경우 이 웹 에이전트는 응용 프로그램 서버 프록시가 설치된 위치에 설치됩니다. 웹 에이전트 옵션 팩은 다른 시스템에 있을 수 있습니다.
- 아티팩트 서비스를 보호하는 정책을 적용합니다. 정책을 적용하려면 아티팩트 서비스에 대한 액세스가 허용된 어설션 당사자 대 신뢰 당사자 파트너 관계를 지정합니다.

에이전트 그룹에 웹 에이전트를 추가하려면 다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "인프라", "에이전트", "에이전트 만들기"를 차례로 선택합니다.
3. 배포에 있는 웹 에이전트의 이름을 지정합니다. "제출"을 클릭합니다.
4. "인프라", "에이전트 그룹"을 차례로 선택합니다.
5. "FederationWebServicesAgentGroup" 항목을 선택합니다.

   "에이전트 그룹" 대화 상자가 열립니다.
6. "추가/제거"를 클릭합니다. 그러면 "에이전트 그룹 구성원" 대화 상자가 열립니다.
7. "사용 가능한 구성원" 목록에서 "선택한 구성원" 목록으로 웹 에이전트를 이동합니다.
8. "확인"을 클릭하여 "에이전트 그룹" 대화 상자로 돌아갑니다.
9. "제출", "닫기"를 차례로 클릭하여 기본 페이지로 돌아갑니다.

검색 서비스를 보호하는 정책을 적용하려면 다음 단계를 수행하십시오.

1. 관리 UI 에서 아티팩트 보호 유형에 대한 레거시 방법을 사용하여 파트너 관계를 구성합니다.
2. 이 파트너 관계를 활성화합니다.
3. "정책", "도메인", "도메인 정책"을 선택합니다.

   사용 가능한 도메인 정책 목록이 표시됩니다.
4. 연필 모양 아이콘을 선택하여 적절한 아티팩트 서비스 정책을 편집합니다.

   SAML 1.1

   FederationWSAssertionRetrievalServicePolicy

   SAML 2.0

   SAML2FWSArtifactResolutionServicePolicy

   참고: 제공된 정책이 기본 정책입니다. 아티팩트 서비스를 보호하기 위해 생성한 모든 정책을 사용할 수 있습니다.
5. "사용자" 탭으로 이동합니다.

   페더레이션 사용자 지정 사용자 저장소가 "사용자 디렉터리" 섹션에 표시됩니다.

6. 수정할 사용자 저장소에 대한 "구성원 추가"를 클릭합니다.

   SAML 1.1

   FederationWSCustomUserStore

   SAML 2.0

   SAML2FederationCustomUserStore

7. 레거시 아티팩트 보호를 구성한 파트너 관계를 선택합니다.

   예:

   - SAML 1.1 파트너 관계의 이름이 Acme 인 경우 affiliate:affiliate:Acme 를 선택
   - SAML 2.0 파트너 관계의 이름이 Demo 인 경우 affiliate:samlsp:Demo 를 선택

8. "확인"을 클릭합니다.

이제 HTTP-아티팩트 싱글 사인온에 대한 파트너 관계에서 아티팩트 서비스에 대한 액세스가 허용되므로 신뢰 당사자가 어설션을 검색할 수 있습니다.

싱글 사인온 구성(신뢰 당사자)

신뢰 당사자에서 싱글 사인온을 구성하려면 SAML 바인딩과 다른 관련 SSO 설정을 지정하십시오.

신뢰 당사자에서 시스템은 파트너 관계에 대한 차이 시간을 사용하여 수신하는 어설션이 유효한지 여부를 확인합니다. 시스템이 구성된 차이 시간을 사용하는 방식을 이해하려면 어설션 유효성 (페이지 131)에 대한 자세한 정보를 읽어 보십시오.

다음 절차에서는 싱글 사인온이 사용되도록 설정하기 위한 기본 단계를 설명합니다. 사인온 대화 상자의 구성 가능한 모든 기능에 대한 자세한 내용은 이후 항목과 관리 UI 도움말에 설명되어 있습니다.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 적절한 단계에서 시작합니다.

   SAML 1.1

   싱글 사인온

   SAML 2.0

   SSO 및 SLO

   WS-페더레이션

   싱글 사인온 및 사인아웃

2. 대화 상자의 SSO 섹션에서 설정을 구성합니다. 이러한 설정으로 싱글 사인온 바인딩을 제어할 수 있습니다.

   필드 설명을 보려면 "도움말"을 클릭하십시오.

   SAML 의 경우 HTTP-아티팩트 또는 HTTP-POST 프로필을 구성하십시오. 신뢰 당사자가 싱글 사인온을 시작하는 경우 신뢰 당사자는 요청에 쿼리 매개 변수를 포함합니다. 이 쿼리 매개 변수는 사용할 SSO 바인딩을 지정합니다. 바인딩을 지정하지 않은 경우 기본값은 POST 입니다. 어설션 당사자가 싱글 사인온을 시작하는 경우 어설션 당사자는 해당 트랜잭션에 사용되는 바인딩을 지정합니다.

3. (선택 사항) SAML 2.0 의 경우 다음 설정을 구성할 수 있습니다.

   - 원격 SSO 서비스 URL
   - 원격 SOAP 아티팩트 URL
   - 싱글 사인온을 시작할 파트너

     타사 IdP 가 호스트에 사용자 레코드가 없는 소비자 사용자를 인증할 경우 SP 에서 SSO 가 시작됩니다.

   - 사용자 동의 요구 사항

4. HTTP-아티팩트 프로필을 선택하는 경우 대화 상자의 "백 채널" 섹션에서 백 채널에 대한 인증 방법을 구성합니다.

5. 나머지 설정의 경우 기본값을 사용합니다.

싱글 사인온의 기본 설정이 완료되었습니다. SSO 에는 다른 설정도 사용할 수 있습니다. 필드 설명을 보려면 "도움말"을 클릭하십시오.

HTTP 오류에 대한 상태 리디렉션(SAML 2.0 IdP)

아이덴티티 공급자의 경우 HTTP 500, 400 또는 405 오류 발생 시 SiteMinder 가 사용자를 리디렉션하는 방법을 구성할 수 있습니다. 예를 들어 요청의 URL 이 잘못된 대상을 가리키기 때문에 403 오류가 발생할 수 있습니다. 이러한 오류가 발생하면 추가 처리를 위해 지정된 URL 로 사용자가 리디렉션됩니다.

다음과 같이 리디렉션 옵션을 선택하십시오.

1. "SSO 및 SLO" 대화 상자의 "상태 리디렉션 URL" 섹션으로 이동합니다.
2. "상태 리디렉션 URL" 섹션에서 리디렉션을 수행하는 오류 조건에 대한 확인란을 선택합니다.
3. SiteMinder 가 사용자를 리디렉션하는 대상 URL 을 입력합니다.
4. 각 URL 에 대해 리디렉션 방법으로 "302 데이터 없음" 또는 "HTTP Post"를 선택합니다.

리디렉션 처리가 구성되었습니다.

싱글 사인온을 시작할 수 있는 SAML 2.0 엔터티

SAML 2.0 파트너 관계의 경우 싱글 사인온을 시작할 수 있는 항목을 IdP 또는 SP 또는 둘 다로 결정할 수 있습니다. 파트너 관계의 각 측에서 허용되는 트랜잭션을 구성할 수 있습니다.

트랜잭션 초기화의 제한이 사용자 인증 컨텍스트 정보 교환과 같은 다른 싱글 사인온 기능에 어떤 영향을 미치는지 고려하십시오.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. 편집할 SAML 2.0 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
4. "트랜잭션 허용됨" 필드의 풀다운 메뉴에서 옵션을 선택합니다.
5. 마법사의 "확인" 단계로 건너뛰고 변경 내용을 저장합니다.

싱글 사인온에 대한 어설션 유효 기간

싱글 사인온의 경우 차이 시간 및 SSO 유효 기간 값에 따라 어설션의 유효 기간을 계산하는 방법이 결정됩니다. 정책 서버는 어설션 생성 및 소비에 이 차이 시간을 적용합니다. 어설션 문서에서 NotBefore 및 NotOnOrAfter 값은 유효 간격의 시작 및 끝을 나타냅니다.

어설션 당사자에서는 정책 서버가 어설션 유효 기간을 설정합니다. 정책 서버는 어설션이 생성될 때 시스템 시간을 가져와서 유효 간격의 시작을 결정합니다. 소프트웨어는 이 시간을 사용하여 어설션의 IssueInstant 값을 설정합니다. 그런 다음 정책 서버가 IssueInstant 값에서 차이 시간 값을 뺍니다. 그 결과로 얻은 시간은 NotBefore 값이 됩니다.

NotBefore = IssueInstant - 차이 시간

유효 간격의 끝을 결정하기 위해 정책 서버는 유효 기간 값과 차이 시간을 IssueInstant 값에 더합니다. 그 결과로 얻은 시간은 NotOnOrAfter 값이 됩니다.

NotOnOrAfter = 유효 기간 + 차이 시간 + IssueInstant

시간은 GMT 를 기준으로 합니다.

예를 들어 어설션 당사자 측에서 어설션이 1:00 GMT 에 생성된다고 가정합니다. 차이 시간은 30 초이고 유효 기간은 60 초이며 어설션 유효 간격은 12:59:30 GMT 에서 1:01:30 GMT 사이입니다. 이 간격은 어설션이 생성된 시간보다 30 초 전에 시작되고 90 초 후에 끝납니다.

신뢰 당사자에서 정책 서버는 어설션 당사자에서와 동일한 계산을 수행하여 수신된 어설션이 유효한지 확인합니다.

SiteMinder 가 파트너 관계의 양쪽 모두에 있는 경우의 어설션 유효 기간 계산

어설션이 유효한 총 시간은 SSO 유효 기간에 차이 시간의 두 배를 더한 합입니다. 공식은 다음과 같습니다.

어설션 유효 기간 = 2 x 차이 시간(어설션 당사자) + SSO 유효 기간 + 2 x 차이 시간(신뢰 당사자)

공식의 처음 부분(2 x 차이 시간 + SSO 유효 기간)은 어설션 당사자의 유효 기간을 나타냅니다. 공식의 두 번째 부분(2 x 차이 시간)은 신뢰 당사자에 있는 시스템 클록의 차이 시간을 나타냅니다. 2 를 곱하는 이유는 유효 기간의 NotBefore 및 NotOnOrAfter 끝을 고려하기 때문입니다.

참고: 정책 서버의 경우 SSO 유효 기간은 어설션 당사자에서만 설정됩니다.

예

어설션 당사자

어설션 당사자 측에서의 값은 다음과 같습니다.

IssueInstant = 5:00PM

SSO 유효 기간 = 60 초

차이 시간 = 60 초

NotBefore = 4:59PM

NotOnOrAfter = 5:02PM

신뢰 당사자

신뢰 당사자는 어설션에서 받은 NotBefore 및 NotOnOrAfter 값을 가져와서 해당 차이 시간을 적용하여 새 값을 계산합니다.

차이 시간 = 180 초(3 분)

NotBefore = 4:56PM

NotOnOrAfter = 5:05PM

이러한 값을 기반으로 총 어설션 유효 기간에 대한 계산은 다음과 같습니다.

120 초(2 x 60) + 60 초 + 360 초(2 x 180) = 540 초(9 분)

서비스 공급자에서의 세션 유효 기간

서비스 공급자에서 인증 세션의 기간을 관리할 수 있습니다. SessionNotOnOrAfter 특성은 IdP 가 어설션의 <AuthnStatement>에 포함할 수 있는 선택적 특성입니다. 어설션 유효 기간의 구성은 IdP 에서 수행됩니다.

참고: SessionNotOnOrAfter 매개 변수는 어설션의 유효 기간을 결정하는 NotOnOrAfter 매개 변수와는 다릅니다.

타사 SP 는 SessionNotOnOrAfter 의 값을 사용하여 자체 시간 만료 값을 설정할 수 있으므로 너무 짧은 세션을 방지하는 데 도움이 됩니다. 사용자 세션이 무효화되면 사용자는 아이덴티티 공급자에서 다시 인증해야 합니다.

중요! SiteMinder 가 SP 로 작동하고 있는 경우 SessionNotOnOrAfter 값이 무시됩니다. 대신에 SiteMinder SP 는 대상 리소스를 보호하는 SAML 인증 체계에 해당하는 영역 시간 만료를 바탕으로 세션 시간 만료를 설정합니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. 수정할 IdP->SP 파트너 관계를 선택합니다.
3. "SSO 및 SLO" 단계로 이동합니다.
4. SSO 섹션에서 "SP 세션 유효 기간"에 대한 옵션을 선택합니다 사용자 지정 옵션을 선택하면 여러 개의 옵션을 선택할 수 있습니다.

   필드 설명을 보려면 "도움말"을 클릭하십시오.
5. 변경을 완료하고 "마침"을 클릭한 후 "확인" 단계를 선택합니다.

아티팩트 SSO 에 대한 백 채널 인증

아티팩트 싱글 사인온을 위해서는 신뢰 당사자가 어설션을 검색하기 위해 어설션 당사자에 아티팩트를 보내야 합니다. 어설션 당사자는 아티팩트를 사용하여 올바른 어설션을 검색하고 백 채널을 통해 어설션을 신뢰 당사자에 반환합니다.

엔터티가 백 채널에 액세스하려면 인증이 필요하도록 설정할 수 있습니다. SSL 은 필수가 아니지만 SSL 을 사용하여 백 채널에 보안을 적용할 수도 있습니다.

SSL 을 사용하여 백 채널에 보안을 적용하는 절차는 다음과 같습니다.

1. SSL 이 사용되도록 설정합니다.

   기본 인증에는 SSL 이 필요하지 않지만 SSL 을 통한 기본 인증을 사용할 수 있습니다. 클라이언트 인증서 인증에는 SSL 이 필요합니다.

2. SAML 2.0 통신 교환에 대한 들어오는 백 채널 또는 나가는 백 채널을 구성합니다. 구성하는 방향은 로컬 엔터티의 역할에 따라 다릅니다.

   별도 채널의 구성은 SAML 2.0 에만 지원됩니다. SAML 1.1 아티팩트 싱글 사인온에 대한 백 채널 구성은 각 파트너 관계에 대해 단일 구성을 사용합니다. SiteMinder 는 자동으로 올바른 방향(로컬 생산자에 대해서는 들어오는 방향, 로컬 소비자에 대해서는 나가는 방향)을 사용합니다.

   구성 중인 엔터티에 기반하여 SAML 2.0 싱글 사인온에 대해 구성할 방향을 선택합니다.

   - 로컬 어설션 당사자는 들어오는 채널을 사용합니다.
   - 로컬 신뢰 당사자는 나가는 채널을 사용합니다.

   참고: 들어오고 나가는 백 채널을 구성할 수 있지만 한 채널은 하나의 구성만 가질 수 있습니다. 동일한 채널을 사용하는 두 서비스는 동일한 백 채널 구성을 사용합니다. 예를 들어 로컬 어설션 당사자의 수신 채널이 HTTP-아티팩트 SSO 와 SOAP 기반 SLO 를 지원할 경우 이 두 서비스는 동일한 백 채널 구성을 사용해야 합니다.

3. 신뢰 당사자가 보호된 백 채널을 통해 액세스를 얻기 위한 인증 유형을 선택합니다. 인증 방법은 채널별로(나가는 채널 또는 들어오는 채널) 적용됩니다.

   백 채널 인증에 대한 옵션은 다음과 같습니다.

   - 기본
   - 클라이언트 인증서
   - 인증 없음

   이러한 옵션은 관리 UI 도움말에 자세히 설명되어 있습니다.

   중요! 들어오는 백 채널에 대한 인증 방법은 파트너 관계에서 다른 측의 나가는 백 채널에 대한 인증 방법과 일치해야 합니다. 인증 방법의 선택에 대한 동의는 대역 외 통신에서 처리됩니다.

SAML 2.0 특성 쿼리 지원

SiteMinder IdP 는 SAML 2.0 어설션 쿼리/요청 프로필을 지원하며 특성 쿼리에 응답할 수 있습니다. 또한 IdP 는 어설션이나 메타데이터에 없는 특성에 대한 쿼리를 수락하여 프로필 기능을 확장합니다. IdP 는 특성 쿼리를 받으면 먼저 사용자 디렉터리에서 특성을 찾습니다. 해당 특성이 없을 경우 정책 서버가 세션 저장소를 확인합니다. 세션 저장소는 외부 아이덴티티 공급자의 특성, 고급 인증 체계에서 수집된 특성 및 다른 원본의 특성을 포함할 수 있습니다.

참고: SiteMinder IdP 만 쿼리 프로필을 지원합니다. 특성 요청자로서 SiteMinder SP 는 프록시된 특성 쿼리 기능 (페이지 138)에 대해서만 지원됩니다.

IdP 에는 SP 가 메타데이터에서 요청할 수 있는 모든 사용자 특성이 있습니다. SP 는 다음과 같은 두 가지 방법으로 이러한 특성을 획득할 수 있습니다.

- 어설션에 전송된 특성 집합을 추출합니다.

  아이덴티티 공급자 어설션 구성은 포함되는 특성 집합을 결정합니다. 모든 특성의 하위 집합을 정의하는 경우 가장 필요한 특성만 포함하도록 특성의 수를 제한하여 처리 오버헤드를 줄일 수 있습니다.

- IdP 메타데이터를 가져옵니다.

메타데이터의 특성에 추가하여, SP 는 어설션 또는 메타데이터에 없는 특성을 요구할 수 있습니다. 다른 특성을 가져오려면 SP 에서 IdP 에 특성 쿼리를 보내야 합니다.

쿼리 요청 프로필은 다음 두 개 엔터티를 갖습니다.

- SAML 특성 기관
- SAML 특성 요청자

SiteMinder IdP 는 특성 기관으로만 기능할 수 있습니다. SiteMinder SP 는 특성 요청자가 될 수 없습니다.

다음 그림은 특성 기관에 대한 구성 단계를 보여 줍니다.

다음 단계를 완료하십시오.

- IdP-SP 파트너 관계를 구성 또는 수정합니다. (페이지 137)
- 아이덴티티 공급자에 있는 경우 SAML 2.0 특성 기관을 구성합니다 (페이지 137).

SiteMinder 가 파트너 관계의 양쪽에 모두 있는 경우 어설션 쿼리/응답 프로필을 사용할 수 없습니다.

특성 쿼리 지원을 위한 파트너 관계 구성

IdP 가 특성 쿼리에 응답하려면 IdP-SP 파트너 관계가 있어야 합니다. 파트너 관계를 만들거나 기존 파트너 관계를 수정할 수 있습니다.

파트너 관계를 만들기 위한 단계는 다음을 포함합니다.

1. SAML 2.0 IdP 및 SP 엔터티를 만듭니다 (페이지 69).
2. 파트너 관계에 대한 사용자 디렉터리로의 연결을 구성합니다 (페이지 63).
3. SAML 2.0 IdP-SP 파트너 관계를 만듭니다 (페이지 81).
4. SAML 2.0 특성 기관을 구성합니다 (페이지 137).

이러한 단계는 이 설명서 내에서 자세히 설명됩니다.

SAML 2.0 특성 기관 구성

IdP 가 특성 기관으로 사용되도록 구성할 수 있습니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.
3. 수정할 IdP-SP 파트너 관계를 선택하거나 새 파트너 관계를 생성합니다.
4. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
5. 대화 상자의 "특성 서비스" 섹션에서 "사용"을 선택합니다.
6. "유효 기간"에 시간(초)을 입력합니다.
7. (선택 사항) 특성 쿼리에 서명이 필요한지 여부와 특성 어설션 및 응답에 대한 서명 요구 사항을 지정합니다.
8. "사용자 조회" 섹션에서 적절한 사용자 디렉터리 네임스페이스에 대한 검색 사양을 입력합니다. 특성 기관은 이 검색 사양을 사용하여 사용자를 명확히 구분합니다.

   LDAP 사용자 디렉터리의 예는 uid=%s 입니다. 하나 이상의 검색 사양이 필요합니다.

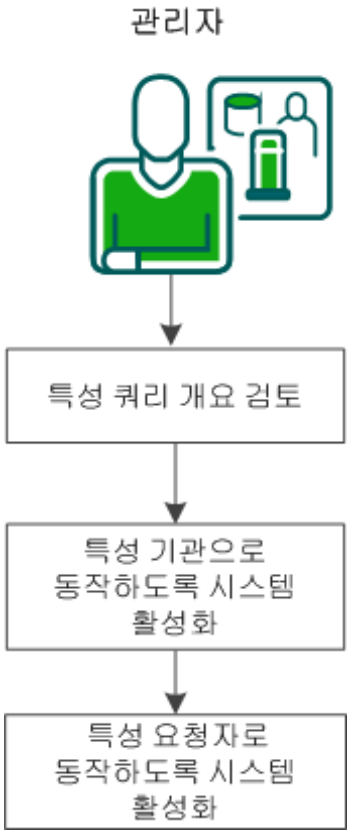
9. (선택 사항) "백 채널" 섹션에서 "보호 유형"으로 "파트너 관계"를 지정합니다. 그런 다음 인증 방법을 선택합니다. 백 채널에 대한 자세한 내용을 보려면 "도움말"을 클릭하십시오.
10. 파트너 관계를 저장한 후 활성화합니다.

이제 아이덴티티 공급자를 특성 기관으로 사용할 수 있습니다. 이 기관은 이제 타사 SP 의 특성 쿼리에 응답합니다.

타사에서 사용자 특성 값을 가져오기(SAML 2.0)

경우에 따라 SAML 2.0 페더레이션된 환경의 서비스 공급자는 어설션에 제공되지 않는 사용자 관련 정보를 필요로 합니다. 서비스 공급자는 미리 결정된 사용자 특성의 값을 요청할 수 있습니다. 아이덴티티 공급자에 이러한 값이 없는 경우 타사에 값을 요청할 수 있습니다. SiteMinder 환경에서는 이 기능을 프록시 특성 쿼리라고 합니다.

다음 다이어그램에서는 프록시 특성 쿼리 기능이 사용되도록 설정하는 프로세스를 보여 줍니다.



프록시 특성 쿼리가 사용되도록 설정하려면 다음 태스크를 완료하십시오.

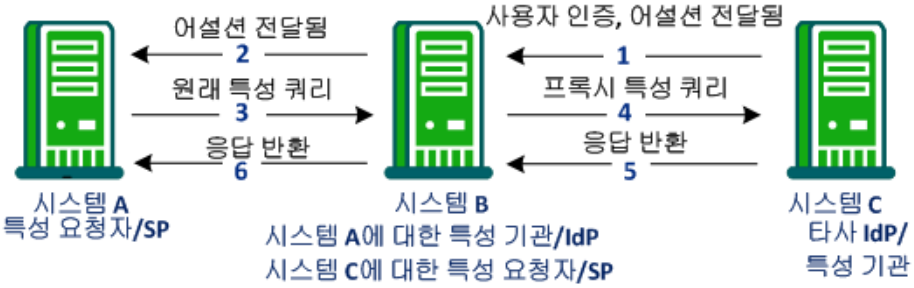
1. 프록시 특성 쿼리 개요를 검토합니다. (페이지 139)
2. 시스템이 특성 기관 역할을 하도록 설정합니다 (페이지 140).
3. 시스템이 특성 요청자 역할을 하도록 설정합니다. (페이지 141)

프록시 특성 쿼리 개요

프록시 특성 쿼리 기능은 SAML 2.0 어설션 쿼리/요청 프로필을 기반으로 하며 사용자 특성 검색을 확장합니다. 특성 기관은 먼저 사용자 디렉터리 및 세션 저장소에서 특성을 검색합니다. 특성을 찾을 수 없고 사용자가 처음에 타사 IdP 에서 인증된 경우 타사 IdP 에 요청을 전달할 수 있습니다.

프록시 특성 쿼리를 구현하기 위해 단일 SiteMinder 시스템은 두 원격 시스템 간의 릴레이 지점으로 사용됩니다. 두 원격 시스템 간에 요청을 릴레이하기 위해 단일 시스템에서 두 가지 역할을 수행합니다. 시스템은 먼저 원래 특성 요청자에 대한 특성 기관으로 사용됩니다. 또한 타사 IdP 에 대한 특성 요청자로 사용됩니다. 특성 요청자로 사용되는 시스템은 특성 쿼리를 원래 IdP 로 프록시합니다.

다음 그림에서는 단일 시스템이 프록시 쿼리를 처리하는 방식을 보여 줍니다.



다음 단계에서는 프록시 특성 쿼리의 흐름을 설명합니다.

1. 처음에 사용자는 타사 IdP 인 시스템 C 에서 인증됩니다. 시스템 C 는 어설션을 생성하여 시스템 B 에 전달합니다.
2. 시스템 B 는 어설션을 시스템 A 로 보내 시스템 A, B, C 사이에서 최초 싱글 사인온 트랜잭션을 완료합니다. 이 싱글 사인온 트랜잭션은 프록시 특성 쿼리를 처리하는 데 필요합니다.

3. 시스템 A 가 어설션을 받은 후에 시스템은 어설션에 없는 다른 특성이 필요함을 파악합니다. 특성 요청자로서 시스템 A 는 특성 기관/IdP, 시스템 B 로 특성 쿼리를 보냅니다.
4. 시스템 B 는 시스템 A 가 사용자 디렉터리 또는 세션 저장소에 없는 특성이 필요함을 파악합니다. 특성을 가져오기 위해 시스템 B 는 새 쿼리 요청을 생성합니다. 새 쿼리를 사용자가 원래 인증된 타사 IdP 인 시스템 C 로 보냅니다. 이 새 쿼리는 프록시 쿼리입니다.
5. 시스템 C 가 특성이 포함된 응답을 시스템 B 에 반환합니다. 시스템 B 가 이러한 특성을 세션 저장소에 저장합니다.
6. 특성 기관 역할을 수행하는 시스템 B 가 특성이 포함된 응답을 시스템 A 에 반환합니다.

중요! 시스템 A 의 구성된 특성 이름 및 이름 형식(지정되지 않음, uri, 기본)은 시스템 C 의 이러한 특성 이름과 일치해야 합니다. 이 정보는 트랜잭션이 발생하기 전에 전달됩니다.

시스템이 특성 기관 역할을 하도록 설정(IdP->SP)

프록시 쿼리 트랜잭션을 구현하려면 동일한 SiteMinder 시스템에서 다음과 같은 두 가지 파트너 관계를 구성하십시오.

- IdP-SP 파트너 관계
- SP-IdP 파트너 관계

SiteMinder 가 특성 기관의 역할을 하기 위해서는 기존 IdP-SP 파트너 관계를 수정하거나 새 파트너 관계를 생성해야 합니다. 이 파트너 관계에서 SiteMinder 는 로컬 IdP/특성 기관이 되고 원격 파트너는 SP/특성 요청자가 됩니다.

참고: 이 시스템은 또한 SP-IdP 파트너 관계에서 특성 요청자로서의 역할을 합니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.
3. 수정할 IdP-SP 파트너 관계를 선택하거나 새 파트너 관계를 생성합니다.
4. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.

5. 대화 상자의 "특성 서비스" 섹션에서 "사용"을 선택합니다.
6. "유효 기간"에 시간(초)을 입력합니다.
7. (선택 사항) 특성 쿼리에 서명이 필요한지 여부와 특성 어설션 및 응답에 대한 서명 요구 사항을 지정합니다.
8. "프록시 쿼리 사용"을 선택합니다.
9. "사용자 조회" 섹션에서 적절한 사용자 디렉터리 네임스페이스에 대한 검색 사양을 입력합니다. 특성 기관은 이 검색 사양을 사용하여 사용자를 명확히 구분합니다.

   LDAP 사용자 디렉터리의 예는 uid=%s 입니다. 하나 이상의 검색 사양이 필요합니다.
10. (선택 사항) "백 채널" 섹션에서 "보호 유형"으로 "파트너 관계"를 지정합니다. 그런 다음 인증 방법을 선택합니다. 백 채널에 대한 자세한 내용을 보려면 "도움말"을 클릭하십시오.
11. 파트너 관계를 저장한 후 활성화합니다.

이제 시스템이 원래 특성 요청자에 대해 특성 기관의 역할을 할 수 있습니다.

시스템이 특성 요청자 역할을 하도록 설정(SP->IdP)

프록시 쿼리 트랜잭션을 구현하려면 동일한 SiteMinder 시스템에서 다음과 같은 두 가지 파트너 관계를 구성하십시오.

- IdP-SP 파트너 관계
- SP-IdP 파트너 관계

참고: 파트너 관계 페더레이션은 프록시 특성 쿼리 기능에 대해서만 특성 요청자로서 SP 를 지원합니다.

SiteMinder 가 특성 요청자의 역할을 하기 위해서는 기존 SP-IdP 파트너 관계를 수정하거나 새 파트너 관계를 생성해야 합니다. 이 파트너 관계에서 SiteMinder 는 로컬 SP/특성 요청자가 되고 원격 타사는 원격 IdP/특성 기관이 됩니다.

참고: 이 시스템은 또한 IdP-SP 파트너 관계에서 특성 기관으로서의 역할도 합니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.
3. 수정할 SP-IdP 파트너 관계를 선택하거나 새 파트너 관계를 생성합니다.
4. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
5. "특성 요청자 서비스" 섹션에서 "사용" 및 "프록시 쿼리 사용"을 선택합니다.
6. "특성 서비스" 섹션에서 원격 IdP 의 URL 을 지정합니다.
7. 이름 ID 의 형식, 유형 및 값을 지정합니다.
8. (선택 사항) 백 채널의 인증 유형을 선택합니다. 백 채널에 대한 자세한 내용을 보려면 "도움말"을 클릭하십시오.
9. 파트너 관계를 저장한 후 활성화합니다.

이제 서비스 공급자를 특성 요청자로 사용할 수 있습니다.

SAML 2.0 IdP 에서 사용자 동의

SiteMinder 아이덴티티 공급자는 SAML 2.0 에 대한 사용자 동의 기능을 지원합니다. 사용자 동의를 사용하려면 아이덴티티 공급자가 파트너에 어설션을 전송하기 전에 사용자에게 권한 부여를 요청해야 합니다. 아이덴티티 공급자에서 사용자 동의가 사용되도록 설정하면 SiteMinder 에서 사용자에게 동의하는지 묻습니다. 아이덴티티 공급자는 어설션에서 동의 값을 전달합니다.

동의 유효 기간은 5 분입니다. 아이덴티티 공급자가 사용자를 동의 페이지로 리디렉션하면 사용자는 동의할 시간을 5 분간 허용 받고 다시 아이덴티티 공급자로 리디렉션됩니다. 그러면 아이덴티티 공급자가 어설션을 생성하여 이를 서비스 공급자로 전송합니다. 이 태스크는 5 분 내에 완료되어야 합니다. 아이덴티티 공급자가 어설션을 생성하기 전에 시간이 만료되면 사용자 아이덴티티를 전달하지 않습니다.

동의는 단일 어설션에만 적용됩니다. 아이덴티티 공급자는 어설션을 생성한 후에 동의가 부여된 모든 레코드를 삭제합니다. 5 분의 유효 기간이 만료되기 전에 동일한 사용자가 아이덴티티 공급자로 돌아갈 수 있지만 아이덴티티 공급자는 여전히 사용자에게 동의할지를 묻습니다.

참고: 유효 기간은 구성할 수 없습니다.

예

User1 이 오후 2 시에 MyWorkPlace.com 에 로그인하고 인증합니다. MyWorkPlace 가 아이덴티티 공급자의 역할을 합니다. 오후 2 시 3 분에 사용자가 직원용 여행 특별 상품을 운영하는 파트너 회사의 링크를 선택합니다. User1 은 ExampleTravel.com 으로 보내지기 전에 동의 여부를 묻는 양식으로 리디렉션됩니다. User1 은 동의 양식을 기입하기 전에 전화 통화를 합니다. 현재 시간은 오후 2 시 10 분입니다. 이 경우 유효 기간이 만료되었기 때문에 MyWorkPlace 는 어설션을 생성하지 않습니다.

User1 이 오후 2 시 5 분까지 신속하게 동의하고 아이덴티티 공급자로 다시 리디렉션되면 아이덴티티 공급자는 어설션을 생성합니다. 동의와 어설션 사이에 2 분만 경과되었으므로 유효 기간이 여전히 활성 상태입니다.

사용자 동의를 구성하려면 다음과 같이 해야 합니다.

- 사용자 동의가 사용되도록 설정합니다.
- 사용자 동의 양식의 이름을 제공합니다.

  아이덴티티 공급자는 사용자의 동의를 구하기 위한 사용자 양식을 전송합니다.

아이덴티티 공급자가 어설션 응답에 사용자 동의 특성을 포함하지 않으면 다음 URI 만 사용됩니다.

```
urn:oasis:names:tc:SAML:2.0:consent:obtained
```

사용자 동의는 서비스 공급자에서도 구성할 수 있습니다. 서비스 공급자는 아이덴티티 공급자에게 어설션 응답에 사용자 동의 값을 전달하도록 요구할 수 있습니다.

사용자 동의 양식 사용자 지정

SiteMinder 에는 ca_defaultconsentform.html 이라는 *페더레이션 동의* 양식이 함께 제공됩니다. 아이덴티티 공급자는 사용자의 동의를 구하기 위한 사용자 양식을 전송합니다. 기본 동의 양식은 %NETE_WA_ROOT%\customization 디렉터리에 있습니다. %NETE_WA_ROOT%는 웹 에이전트 옵션 팩의 위치입니다.

기본 동의 양식을 사용하고 관리 UI 에서 양식을 지정하는 대신 사용자 지정 양식을 사용할 수 있습니다.

다음 단계를 수행하십시오.

1. 사용자 지정 HTML 양식을 생성합니다. 양식을 수정하고 다음 설정에 대한 값을 바꿉니다.

   $$userconsent_spid$$

   파트너 관계에서 구성된 SP ID 를 나타냅니다.

   $$userconsent_idpid$$

   파트너 관계에서 구성된 IDP ID 를 나타냅니다.

2. 양식을 %NETE_WA_ROOT%\customization 디렉터리에 넣습니다.

   NETE_WA_ROOT 는 시스템 환경 변수입니다. %NETE_WA_ROOT%는 웹 에이전트 옵션 팩의 위치입니다. 웹 에이전트와 웹 에이전트 옵션 팩을 동일한 시스템에 설치하는 경우 동일한 디렉터리(예: webagent\customization)에 설치됩니다.

3. 관리 UI 에 로그인합니다.
4. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"로 이동합니다.
5. 수정할 IdP->SP 파트너 관계를 선택합니다.
6. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
7. SSO 섹션에서 다음을 수행합니다.
   a. "사용자 동의 사용" 확인란을 선택합니다.
   b. "사용자 동의 Post 양식" 필드에서 사용자 지정 양식의 이름을 지정합니다.

   참고: "사용자 동의 서비스 URL"은 기본적으로 지정됩니다. 이 값은 변경할 수 없습니다.
8. 구성이 완료되면 확인 단계로 이동하고 "마침"을 클릭합니다.

ECP(향상된 클라이언트 또는 프록시) 프로필 개요(SAML 2.0)

ECP(향상된 클라이언트 또는 프록시) 프로필은 싱글 사인온을 위한 응용 프로그램입니다. 향상된 클라이언트는 ECP 기능을 지원하는 브라우저 또는 일부 다른 사용자 에이전트입니다. 향상된 프록시는 무선 장치용 무선 액세스 프로토콜 프록시와 같은 HTTP 프록시입니다.

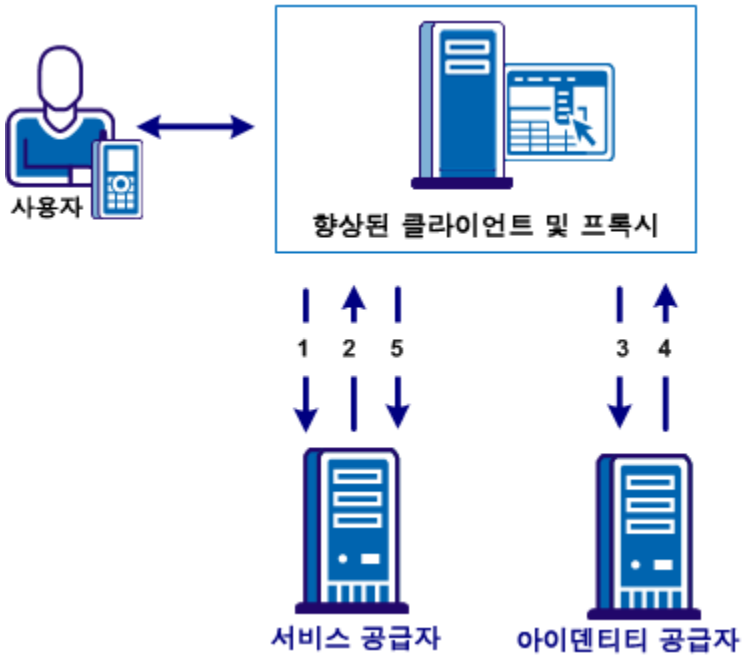
ECP 프로필은 아이덴티티 공급자와 서비스 공급자가 직접 통신할 수 없을 때 싱글 사인온을 가능하게 합니다. ECP 는 서비스 공급자와 아이덴티티 공급자 사이에서 중재자 역할을 합니다.

중재자 역할을 하는 것 외에도 ECP 프로필은 다음과 같은 경우에 유용합니다.

- 이 프로필이 필요한 향상된 클라이언트 또는 프록시에 서비스를 제공할 것으로 예상되는 서비스 공급자의 경우
- 기능이 제한된 모바일 장치 앞의 WAP(무선 액세스 프로토콜) 게이트웨이와 같은 프록시 서버가 사용되고 있는 경우

ECP 응용 프로그램은 사용자가 직접 얻거나 개발해야 합니다. SiteMinder 는 SAML 요구 사항에 맞는 ECP 응용 프로그램에 대한 ECP 요청 및 응답만 처리합니다.

다음 그림에서는 ECP 프로필의 흐름을 보여 줍니다.



ECP 통신에서 사용자는 휴대폰 등에서 응용 프로그램에 대한 액세스를 요청합니다. 응용 프로그램은 서비스 공급자에 있으며 사용자에 대한 아이덴티티 정보는 아이덴티티 공급자에 있습니다. 서비스 공급자와 아이덴티티 공급자는 직접 통신하지 않습니다.

이 호출의 흐름은 다음과 같습니다.

1. ECP 응용 프로그램이 리버스 SOAP(PAOS) 요청을 서비스 공급자에 전달합니다. 서비스 공급자는 아이덴티티 공급자에 직접 액세스할 수 없습니다.

   아이덴티티 공급자와 달리 ECP 엔터티에는 항상 직접 액세스할 수 있습니다.
2. 서비스 공급자는 ECP 응용 프로그램에 AuthnRequest 를 되보냅니다.
3. ECP 응용 프로그램은 AuthnRequest 를 처리 및 수정하여 아이덴티티 공급자에 보냅니다.
4. 아이덴티티 공급자는 요청을 처리한 후 ECP 응용 프로그램에 SOAP 응답을 반환합니다. 이 응답에는 어설션이 포함됩니다.
5. ECP 응용 프로그램은 서명된 PAOS 응답을 서비스 공급자에 되보냅니다.

싱글 사인온이 진행되고 사용자가 응용 프로그램에 대한 액세스 권한을 얻습니다.

아이덴티티 공급자에서 ECP 구성

ECP 를 구성하려면 아이덴티티 공급자와 서비스 공급자에서 해당 기능을 사용하도록 설정하십시오. 다음 절차는 SiteMinder 아이덴티티 공급자에 해당됩니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. 수정할 로컬 아이덴티티 공급자 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.

4. "SSO" 섹션에서 "향상된 클라이언트 또는 프록시 프로필 사용" 확인란을 선택합니다.
5. "확인" 단계로 이동하고 "마침"을 클릭하여 변경 내용을 저장합니다.

이제 아이덴티티 공급자가 ECP 호출을 처리할 수 있습니다.

참고: 단일 서비스 공급자 개체가 싱글 사인온 요청에 대한 아티팩트, POST, SOAP 및 PAOS 바인딩을 처리할 수 있습니다. SOAP 및 PAOS 는 ECP 프로필에 대한 바인딩입니다. 아이덴티티 공급자와 서비스 공급자는 요청의 매개 변수를 기준으로 사용되는 바인딩을 확인합니다.

서비스 공급자에서 ECP 구성

ECP 를 구성하려면 아이덴티티 공급자와 서비스 공급자에서 해당 기능을 사용하도록 설정해야 합니다. 다음 절차는 서비스 공급자에 해당됩니다.

다음 단계를 수행하십시오.

1. 서비스 공급자에서 보호된 리소스에 대한 요청을 AuthnRequest 서비스에 전달합니다. URL 의 예는 다음과 같습니다.

   https://*host*:*port*/affwebservices/public/saml2authnrequest
2. 관리 UI 에 로그인합니다.
3. 관련된 로컬 서비스 공급자 파트너 관계를 수정합니다.
4. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
5. "SSO" 섹션에서 "향상된 클라이언트 또는 프록시 프로필 사용" 확인란을 선택합니다.
6. "확인" 단계로 이동하고 "마침"을 클릭하여 변경 내용을 저장합니다.

이제 서비스 공급자가 ECP 호출을 처리할 수 있습니다.

참고: 단일 서비스 공급자 개체가 싱글 사인온 요청에 대한 아티팩트, POST, SOAP 및 PAOS 바인딩을 처리할 수 있습니다. SOAP 및 PAOS 는 ECP 프로필에 대한 바인딩입니다. 아이덴티티 공급자와 서비스 공급자는 요청의 매개 변수를 기준으로 사용되는 바인딩을 확인합니다.

IDP 검색 프로필(SAML 2.0)

IPD(아이덴티티 공급자 검색) 프로필이 제공하는 공통 검색 서비스를 사용하면 서비스 공급자가 인증을 위해 고유 IdP 를 선택할 수 있습니다. 네트워크에 있는 모든 사이트가 아이덴티티 공급자 검색 서비스와 상호 작용하도록 파트너 간의 사전 비즈니스 계약이 설정되어 있습니다.

이 프로필은 어설션을 제공하는 파트너가 둘 이상 있는 페더레이션된 네트워크에 유용합니다. 서비스 공급자는 자신이 특정 사용자에 대한 인증 요청을 보내는 아이덴티티 공급자를 결정할 수 있습니다.

IdP 검색 프로필은 두 페더레이션된 파트너에 공통적인 쿠키 도메인을 사용하여 구현됩니다. 약정된 도메인의 쿠키에는 사용자가 방문한 IdP 목록이 포함되어 있습니다.

아이덴티티 공급자의 IDP 검색 구성

"SSO 및 SLO" 대화 상자의 "IDP 검색 섹션"에서 IDP 검색 프로필을 구성하십시오.

참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

다음 단계를 수행하십시오.

1. "IDP 검색 사용" 확인란을 선택합니다.
2. "서비스 URL" 필드의 값을 "아이덴티티 공급자 검색 프로필" 서블릿으로 설정합니다. SiteMinder 의 경우 이 URL 은 다음과 같습니다.

   http://*host*:*port*/affwebservices/public/saml2ipd

   host

   "일반 도메인" 필드에 지정하는 일반 도메인을 나타냅니다.

   port

   제품을 설치할 때 지정한 Apache HTTP 또는 HTTPS 포트를 지정합니다.

   URL 은 https 로 시작할 수도 있습니다.

3. "일반 도메인" 필드에 쿠키 도메인을 지정합니다.
4. (선택 사항) 브라우저에 영구 쿠키를 유지하려면 "영구 쿠키 사용" 확인란을 선택합니다.

IdP 에서 IdP 검색이 활성화됩니다.

서비스 공급자의 IDP 검색 구성

IDP 검색 프로필의 경우 SP(서비스 공급자)는 인증 요청을 보낼 IdP(아이덴티티 공급자)를 확인해야 합니다. SP 가 인증하려는 사용자는 이전에 아이덴티티 공급자를 방문하여 인증을 받은 상태여야 합니다.

일반 도메인 쿠키를 검색하려면 SP 가 사용자를 자체의 IdP 검색 서비스로 리디렉션해야 합니다. 쿠키에는 사용자가 이미 방문한 아이덴티티 공급자의 목록이 포함됩니다. 쿠키는 이 목록에서 올바른 IdP 를 선택한 다음 이 IdP 에 AuthnRequest 를 전송합니다.

IDP 검색 프로세스는 다음과 같습니다.

1. 브라우저가 SP 의 사이트 선택 페이지를 요청합니다.

   이 사이트 선택 페이지는 IDP 검색 서비스 URL 을 인식하고 있습니다.
2. 사이트 선택 페이지는 사용자를 IDP 검색 서비스 URL 로 리디렉션하고 일반 도메인 쿠키를 가져오고자 한다고 표시합니다.
3. IDP 검색 서비스는 일반 도메인 쿠키를 가져오고, 해당 도메인에서 쿠키를 읽고, 사용자를 다시 사이트 선택 페이지로 리디렉션합니다. 검색 서비스는 일반 도메인 쿠키를 쿼리 매개 변수로 제공합니다.
4. SP 는 사이트 선택 페이지에 사용자가 이전에 인증한 IdP URL 을 입력합니다.
5. 사용자가 IdP 를 선택하여 사용자 인증을 수행합니다.

SP 에서 IdP 검색을 구성하려면

1. SP 의 IdP 검색 서비스에서 일반 도메인 쿠키를 요청하는 사이트 선택 페이지를 생성합니다.

   SiteMinder 는 SP 가 IdP 검색을 구현하는 데 사용할 수 있는 IdpDiscovery.jsp 라는 샘플 사이트 선택 페이지와 함께 제공됩니다. 이 페이지는 다음 디렉터리에서 찾을 수 있습니다.

   web_agent_home/affwebservices/public

   첫 번째 링크는 브라우저를 한 도메인에서 일반 도메인의 IdPDiscovery 서비스로 리디렉션하고 **_saml_idp** 라는 일반 도메인 쿠키를 검색합니다. SP 의 IdP 검색 서비스는 요청을 수신하면 일반 도메인 쿠키를 가져와서 이를 쿼리 매개 변수로 추가합니다. 그런 다음 IDP 검색 서비스가 사용자를 일반 도메인의 IdPDiscovery.jsp 사이트 선택 페이지로 다시 리디렉션합니다. 기본적으로 IdPDiscovery.jsp 페이지에는 공용 쿠키에서 추출하는 IdP 에 대한 ID 목록만 표시됩니다. 이 목록은 정적이고, 연결된 IdP 와의 통신을 시작하는 목록과 연결된 HTML 링크가 없습니다.

2. SP 사이트에 대한 샘플 페이지에서 다음 링크를 편집합니다. 링크의 첫 번째 부분에서는 saml2idp 쿠키가 있는 일반 도메인을 지정합니다. 링크의 두 번째 부분에서는 IdPDiscovery.jsp 가 있는 일반 도메인을 지정합니다.

   예를 들면 다음과 같습니다.

   ```
   <a href="http://myspsystem.commondomain.com/affwebservices/public
   /saml2ipd/?IPDTarget=/http://myspsystem.spdomain.com/affwebservices
   /public/IdpDiscovery.jsp&SAMLRequest=getIPDCookie">
   Retrieve idp discovery cookie from IPD Service</a>
   ```

   사용자가 대상 사이트 선택 페이지가 있는 일반 도메인으로 다시 리디렉션되면 이제 해당 사용자가 공용 쿠키를 갖게 됩니다.

3. (선택 사항) 각 IdP 에 대해 HTML 링크를 표시하도록 IdPDiscovery.jsp 사이트 선택 페이지를 편집합니다. 각 링크는 싱글 사인온을 시작하도록 IdP 에 대한 AuthNRequest 를 트리거합니다. 기본적으로 IdPDiscovery.jsp 페이지에는 공용 쿠키에서 추출하는 IdP 에 대한 ID 목록만 표시됩니다.

4. 편집된 사이트 선택 페이지를 사용하여 IdP 검색을 테스트합니다.

IdP 검색이 제대로 작동하는 경우 사이트 선택 페이지에는 선택할 IdP 목록이 표시됩니다.

Office 365 에 대한 싱글 사인온

CA SiteMinder?Federation 는 엔터프라이즈 사용자와 Office 365 서비스 사이에서 싱글 사인온을 사용할 수 있게 해 줍니다. Office 365 로 페더레이션하면 로컬에 서비스를 호스팅해야 하는 부담이 없습니다. 예를 들어, 엔터프라이즈 사용자는 데스크톱 전자 메일 클라이언트 서비스가 클라우드에서 호스트되는 것을 알지 못하고 클라이언트에 로그인합니다. 온-프레미스 응용 프로그램에 연결된 것처럼 Office 365 로그인이 동일하게 이루어집니다.

다음 프로필은 Office 365 에 대한 싱글 사인온에 사용할 수 있습니다.

WS-페더레이션 피동 요청자 프로필

WS-페더레이션 피동 요청자 프로필은 주로 웹 브라우저 또는 HTTP 를 지원하는 브라우저 기반 응용 프로그램과 같은 피동 요청자에 사용할 수 있습니다. 피동 프로필은 이러한 클라이언트와 Microsoft Office 365 사이에서 싱글 사인온을 사용할 수 있게 해 줍니다.

WS-페더레이션 능동 요청자 프로필

보안 토큰 서비스(STS)는 WS-페더레이션 능동 요청자 프로필을 구현합니다. 이 프로필은 SOAP 기반 데스크톱 클라이언트와 다음 Office 365 서비스 사이에서 싱글 사인온을 사용할 수 있게 해 줍니다.

- Exchange Online(Outlook)
- Lync Online
- Dynamics CRM Online

클라이언트는 HTTP-POST 요청 및 응답을 사용하여 SOAP 메시지를 보내고 받습니다. 사용자는 자신의 엔터프라이즈 자격 증명을 사용하여 로그인하고 Outlook 및 Lync 에 액세스할 수 있습니다.

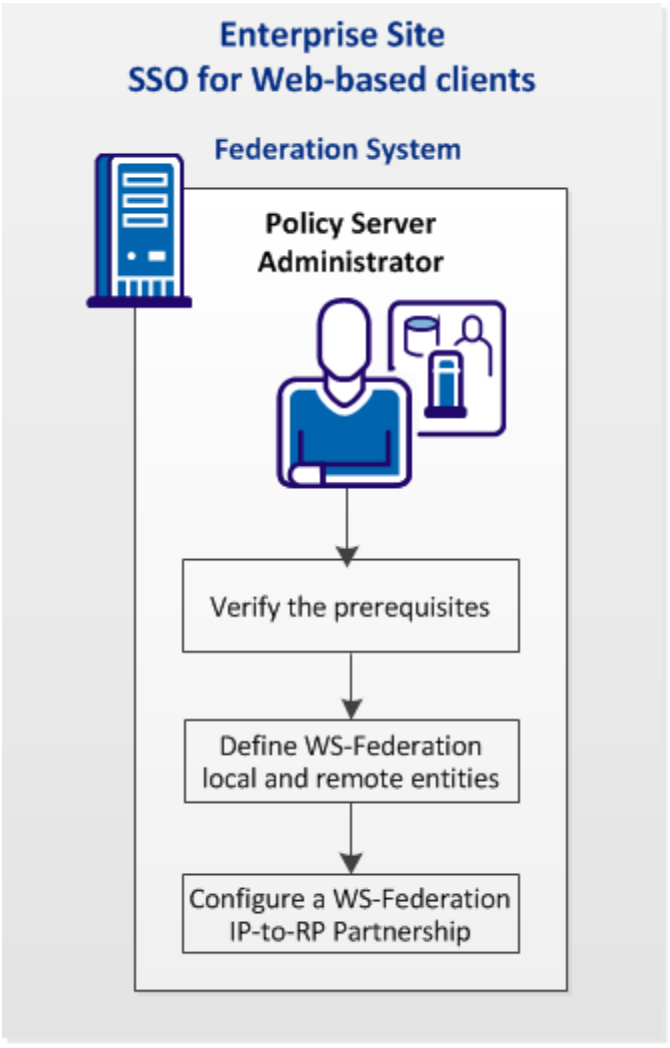
CA SiteMinder® Federation 는 Office 365 가 신뢰하는 아이덴티티 공급자로서 기능하는 STS 서비스를 제공합니다. STS 서비스는 Office 365 서비스가 소비할 수 있는 보안 토큰을 발생합니다.

Office 365 에 싱글 사인온을 구현하려면 두 WS-페더레이션 프로필 모두에 대해 WS-페더레이션 IP 에서 RP 로의 파트너 관계를 페더레이션 시스템에서 구성해야 합니다.

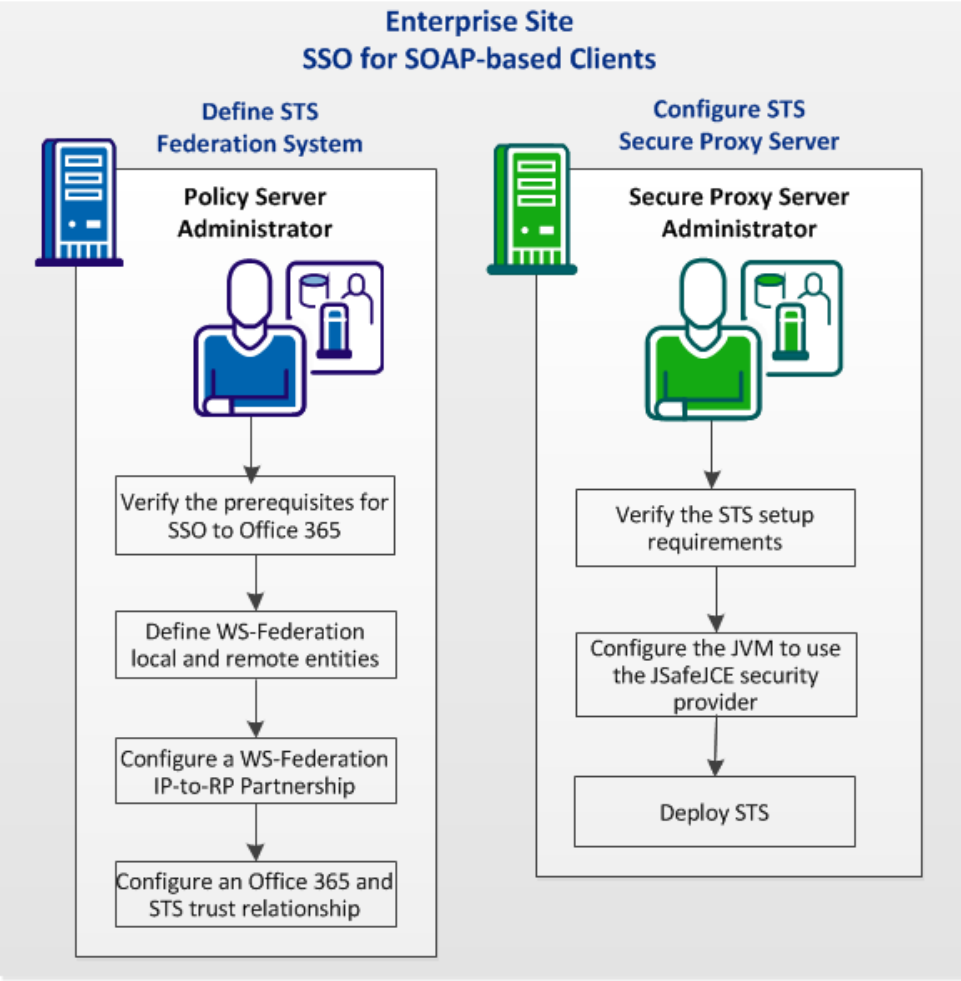
WS-페더레이션 능동 요청자 프로필의 경우 다음 추가 구성 요소도 필요합니다.

- 파트너 관계에서 활성화된 STS 서비스
- SiteMinder 보안 프록시 서버(SPS)에서 구성된 STS

다음 그림은 웹 기반 클라이언트 SSO(피동 요청자 프로필)에 대해 필요한 구성 단계를 보여 줍니다.



다음 그림은 SOAP 기반 클라이언트 SSO(능동 요청자 프로필)에 대해 필요한 구성 단계를 보여 줍니다.



페더레이션 시스템에서 다음 태스크를 완료하십시오.

1. Office 365 에 대한 SSO 의 사전 요구 사항을 확인합니다 (페이지 154).
2. WS-페더레이션 로컬 IP (페이지 157) 및 원격 RP (페이지 160) 항목을 정의합니다.
3. WS-페더레이션 IP-RP 파트너 관계를 구성합니다 (페이지 161).
4. Office 365 와 STS 사이에서 트러스트 관계를 구성합니다. (페이지 164) (SOAP-기반 SSO 만 해당)

STS 를 배포하려면 CA SiteMinder for Secure Proxy Server 에서 다음 태스크를 수행하십시오(SOAP 기반 SSO 만 해당).

1. STS 설정 요구 사항을 확인합니다 (페이지 166).
2. JSafeJCE 보안 공급자를 사용하도록 JVM 을 구성합니다 (페이지 166).
3. CA SiteMinder for Secure Proxy Server 에 STS 을 배포합니다 (페이지 167).

Office 365 와의 싱글 사인온에 대한 자세한 내용은 SiteMinder Federation Cloud Runbook Library (SiteMinder 페더레이션 클라우드 런북 라이브러리)에서 해당 런북을 참조하십시오. 이 콘텐츠를 보려면 로그인해야 합니다.

Office 365 에 대한 SSO 의 사전 요구 사항 확인

Office 365 에 대한 싱글 사인온을 사용하려면 다음과 관련된 요구 사항이 있습니다.

- Office 365 설정
- SiteMinder 사용자 디렉터리

Office 365 설정 요구 사항

- Office 365 도메인을 등록 및 획득합니다. 등록하는 플랜은 싱글 사인온을 지원해야 합니다.
- 소유한 도메인을 등록합니다.
- Office 365 도메인에 도메인을 추가합니다.
- 소유한 Office 365 도메인에 대한 DNS 레코드를 업데이트합니다.

Office 365 에서 작업하기 위해 배포를 구성하는 방법에 대한 자세한 내용은 관련 Microsoft 설명서의 지침을 참조하십시오.

- Office 365 를 등록 및 구독합니다.
- 온-프레미스 및 Office 365 사용자 디렉터리 사이에서 디렉터리 동기화를 구성합니다.
- STS 를 사용하여 구성된 온-프레미스 보안 프록시 서버와 Office 365 와 트러스트 관계를 설정합니다.

Office 365 와의 싱글 사인온에 대한 자세한 내용은 SiteMinder Federation Cloud Runbook Library (SiteMinder 페더레이션 클라우드 런북 라이브러리)에서 해당 런북을 참조하십시오. 이 콘텐츠를 보려면 로그인해야 합니다.

SiteMinder 사용자 디렉터리 요구 사항

- 관리 UI 에서 사용자 디렉터리 연결을 구성할 때는 페더레이션 사용자에 대해 ImmutableID 및 UPN 특성이 있는지 확인하십시오. 온-프레미스 사용자 디렉터리의 이러한 특성에 대한 값은 Office 365 디렉터리에 있는 값과 일치해야 합니다.

  Immutable ID 및 UPN 이 필요합니다. WS-페더레이션 파트너 관계가 필요합니다.

추가 정보:

STS 설정 요구 사항을 확인합니다. (페이지 166)

Office 365 와 WS-페더레이션 파트너 관계 구성

Office 365 와 WS-페더레이션 파트너 관계를 구성합니다. 웹 기반 또는 SOAP 기반 클라이언트 SSO 에는 WS-페더레이션 IP 에서 RP 로의 파트너 관계가 필요합니다.

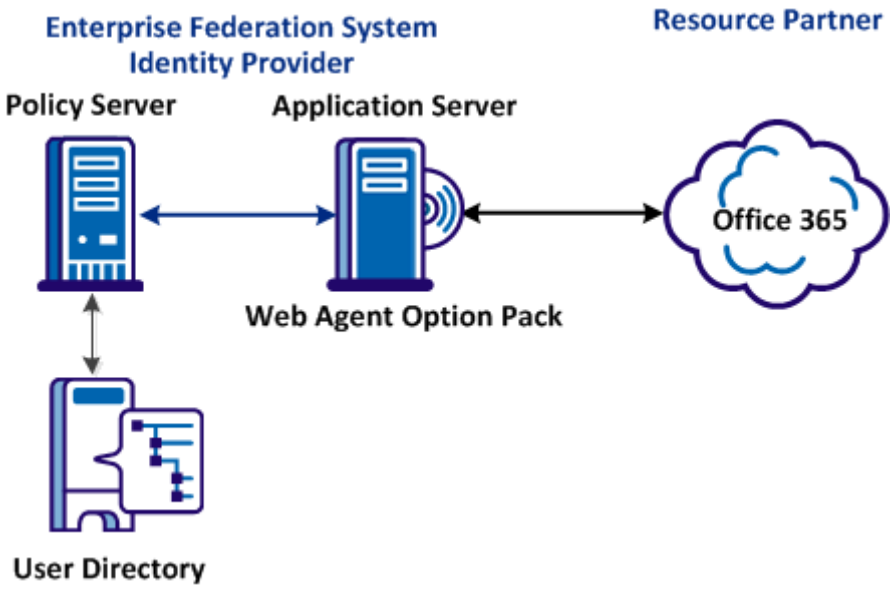
이 파트너 관계에서:

- SiteMinder 는 아이덴티티 공급자(IP)입니다.
- Office 365 는 리소스 파트너(RP)입니다.

WS-페더레이션 피동 요청자 프로필과 능동 요청자 프로필을 구성하기 위한 이 파트너 관계 사이의 차이점은 다음과 같습니다.

- 능동 요청자 프로필에 대한 STS 활성화
- 사인아웃을 구성합니다(선택 사항). 사인아웃은 WS-페더레이션 피동 요청자 프로필에만 해당됩니다.

다음 그림은 이 페더레이션된 솔루션에 대해 권장되는 배포를 보여 줍니다.



Office 365 파트너 관계를 위한 로컬 IP 엔터티 정의

온-프레미스 페더레이션 시스템은 Office 365 과의 파트너 관계에서 아이덴티티 공급자가 됩니다. 아이덴티티 공급자로서 이 시스템은 SAML 1.1 어설션을 수록한 보안 토큰을 발급합니다.

SAML 1.1 토큰을 사용하여 아이덴티티 공급자 엔터티를 만드십시오.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "페더레이션", "파트너 관계 페더레이션", "엔터티"를 선택합니다.
3. "엔터티 만들기"를 클릭합니다.

   "엔터티 만들기" 대화 상자가 표시됩니다.
4. 사이트에 대하여 로컬인 엔터티를 생성하는 것임을 나타내기 위해 **Local** 을 선택합니다.
5. 나머지 필드를 구성합니다.

   새 엔터티 유형

   WSFED 아이덴티티 공급자를 선택합니다.

   SAML 토큰 유형

   SAML 1.1
6. "다음"을 클릭하여 엔터티의 세부 사항을 구성합니다.

   "엔터티 구성" 단계에서 대화 상자의 모든 필수 필드를 입력합니다. 다음 필드는 특히 주의하십시오.

   엔터티 ID

   Office 365 도메인에 지정된 IssuerURI 를 입력합니다.

   이 로컬 파트너에 대해 엔터티 ID 는 고유할 필요가 없습니다.

   엔터티 이름

   이 로컬 IP 를 식별하는 임의의 이름을 입력하십시오. "엔터티 이름"은 정책 저장소에 있는 엔터티 개체를 식별하며 고유한 값을 가져야 합니다. 이 값은 내부용으로만 사용되고 원격 파트너는 이 값을 알지 못합니다.

기준 URL

시스템에 대한 URL 을 지정하십시오. Office 365 와의 통신을 위해 이 URL 은 SSL 연결을 사용해야 합니다. 예: https://fedserver.example.com

명확성 ID(Office 365 에 필요)

동일한 IP 또는 RP 사이에 여러 파트너 관계가 있고 회사에서 Office 365 와의 별도 관계가 있는 사업부가 있는 경우에만 이 ID 를 설정하십시오. Office 365 는 RP 로서 자신을 식별하기 위한 단일 ID 를 사용합니다. SiteMinder Federation 은 동일한 IP 또는 RP ID 를 사용한 여러 파트너 관계를 허용하지 않습니다. 명확성 ID 는 시스템이 특정 파트너에 지정된 서비스 URL 에 대한 고유 논리 경로 접미사를 사용하여 파트너 관계를 구분할 수 있도록 해 줍니다. 하나의 페더레이션 서비스만 있지만 RP ID 와 결합된 접미사는 고유 파트너 관계 조회 키를 생성합니다.

예: microsoftonline

"명확성 ID"는 요청이 올바른 원격 파트너로 전달될 수 있도록 페더레이션 서비스 URL 에 추가됩니다.

예:

능동 요청자 서비스 URL:
https://fedserver1.forwardinc.com/affwebservices/public/wsfeddispatcher/microsoftonline

"microsoftonline"은 명확성 ID 입니다.

영숫자 문자열을 입력하되 특수 문자는 사용하지 마십시오.

사인아웃 확인 URL

사인아웃을 수행하는 아이덴티티 공급자의 URL 을 지정합니다.

기본값: http://*ip_server:port*/affwebservices/signoutconfirmurl.jsp

ip_server:port

아이덴티티 공급자 시스템의 서버 및 포트 번호를 지정합니다. 시스템은 페더레이션 네트워크에 설치된 구성 요소에 따라 웹 에이전트 옵션 팩이나 SPS 페더레이션 게이트웨이를 호스트하고 있습니다.

서명 개인 키 별칭

서명 및 암호화 기능을 사용하려면 인증서 데이터 저장소로 적절한 키 및 인증서 쌍을 가져오십시오.

중요! 이 개인 키와 관련된 공용 인증서는 Office 365 페더레이션 도메인으로 가져와야 합니다.

지원되는 이름 ID 형식 및 특성

지정되지 않음

UPN 특성

UPN 은 사용자 프린서펄 이름입니다.

어설션 특성

UPN

네임스페이스

http://schemas.xmlsoap.org/claims

참고: 이 값은 Microsoft 가 제공합니다. 표시된 대로 네임스페이스 값을 입력하십시오. Office 365 는 이 정확한 값이 필요합니다.

Immutable ID 특성

ImmutableID 는 온-프레미스 Microsoft 디렉터리에서 사용자를 구분하는 고유 특성입니다.

어설션 특성

ImmutableID

네임스페이스

http://schemas.microsoft.com/LIVEID/Federation/2008/05

참고: 이 값은 Microsoft 가 제공합니다. 표시된 대로 네임스페이스 값을 입력하십시오. Office 365 는 이 정확한 값이 필요합니다.

7. 모든 필수 필드가 완성되었으면 "확인"을 클릭하십시오.
8. 원격 엔터티를 구성하십시오.

Office 365 파트너 관계를 위한 원격 RP 엔터티 정의

Office 365 를 나타내는 원격 리소스 파트너를 만드십시오. 엔터티를 정의하려면 메타데이터(있는 경우)를 가져오거나 다음 절차를 따라 엔터티를 구성하십시오.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "페더레이션", "파트너 관계 페더레이션", "엔터티"를 선택합니다.
3. "엔터티 만들기"를 클릭합니다.

   "엔터티 만들기" 대화 상자가 표시됩니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
4. 위치에 대하여 원격인 엔터티를 생성하는 것임을 나타내기 위해 Remote 를 선택합니다.
5. 나머지 필드를 구성합니다.

   새 엔터티 유형

   WSFED 리소스 파트너를 선택합니다.

   SAML 토큰 유형

   SAML 1.1
6. "다음"을 클릭하여 엔터티의 세부 사항을 구성합니다.
7. "엔터티 구성" 단계에서 다음 필수 필드를 완성합니다.

   엔터티 ID

   urn.federation:MicrosoftOnline.

   엔터티 이름

   RP 를 식별하는 임의의 이름을 입력합니다.

원격 보안 토큰 소비자 서비스 URL

Office 365 보안 토큰 서비스에 대한 URL 을 지정합니다. 이 URL 은 Microsoft 가 제공합니다. 이 URL 은 SSL 연결을 사용해야 합니다. 예: https://login.microsoftonline.com

원격 사인아웃 URL(피동 요청자 프로필만 해당)

Office 365 사인아웃 서비스에 대한 URL 을 지정합니다. 이 URL 은 Microsoft 가 제공합니다. 이 URL 은 SSL 연결을 사용해야 합니다. 예: https://login.microsoftonline.com

참고: Office 365 의 경우 보안 토큰 소비자 서비스의 URL 과 사인아웃 URL 이 동일해야 합니다.

지원되는 이름 ID 형식

지정되지 않음

8. 구성을 확인한 후 "확인"을 클릭합니다.

Office 365 와 WS-페더레이션 파트너 관계 구성

로컬 IP 및 원격 RP 엔터티를 만든 후에는 WS-페더레이션 파트너 관계를 구성하십시오. 하나 또는 다른 WS-페더레이션 프로필에 관련된 단계가 설명되어 있습니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그온합니다.
2. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"로 이동합니다.
3. "파트너 관계 만들기" 풀다운 메뉴에서 "WSFED IP->RP"를 선택합니다.

   대화 상자가 열리고 맨 위에 파트너 관계 마법사가 표시됩니다.
4. 파트너 관계 마법사의 1 단계에서 표준 페더레이션 구성에 대한 필수 필드를 완성합니다.
   a. 능동 요청자 프로필에 대해서만 **STS for WSFED Active Profile**(WSFED 능동 프로필용 STS) 확인란을 선택합니다.

5. 필요한 경우, 능동 프로필 끝점 및 데이터를 반영하는 페더레이션 메타데이터 문서를 만들기 위해 "메타데이터 교환 사용"을 선택합니다. 이 문서의 URL 은 다음과 같습니다.

   https://*sps_host*/affwebservices/public/FederationMetadata/*partnership_name*

   참고: *sps_host* 는 STS 가 구성된 보안 프록시 서버입니다.

   메타데이터 문서는 WS-페더레이션 피동/능동 프로필 끝점 및 데이터 등과 같은 파트너 관계에 대한 세부 정보를 제공합니다.

6. 파트너 관계 마법사의 2 단계에서 이 파트너 관계에 대한 페더레이션 사용자를 선택합니다.

7. 파트너 관계 마법사의 3 단계에서 "이름 ID"에 대한 필수 설정을 입력합니다.

   이름 ID 형식

   지정되지 않음

   이름 ID 유형

   사용자 특성

   이름 ID 값

   Office 365 에서 할당된 Immutable ID

8. 마법사의 3 단계에서 "어설션 특성" 설정을 완성합니다.

   a. UPN 및 ImmutableID 어설션 특성이 로컬 IP 엔터티 (페이지 157)에서 상속되었는지 확인합니다. 엔터티 수준에서 이러한 특성을 추가하지 않았으면 여기서 지정합니다.

   b. "유형" 필드를 두 특성에 대한 "사용자 특성"으로 설정합니다.

   c. "값" 필드를 각각 UPN 및 ImmutableID 값을 갖는 사용자 디렉터리 특성으로 설정합니다.

9. 파트너 관계 마법사의 4 단계 "인증" 섹션에서 다음 필드를 완성합니다.

   인증 모드

   로컬

   인증 URL

   WS-페더레이션 피동 요청자 프로필을 사용하는 경우 이 필드를 완성하십시오. "능동 요청자 프로필"에 대해서는 이 필드를 무시하십시오.

   https://*web_agent_optionpack_system*/affwebservices/redirectjsp/redirect.jsp

10. 마법사의 4 단계의 SSO 섹션 및 SLO 섹션에서 다음 필드를 완성합니다.

    대상자

    urn:federation:MicrosoftOnline

    보안 토큰 소비자 서비스 URL

    https://login.microsoftonline.com/

    WS-페더레이션 피동 요청자 프로필에 대해서만 사인아웃 필드를 완성하십시오. 이러한 필드는 능동 요청자 프로필에 해당되지 않습니다.

    사인아웃 확인 URL(선택 사항)

    사인아웃이 구성된 경우 배포를 위한 URL 을 입력합니다.

    사인아웃 URL(선택 사항)

    https://login.microsoftonline.com/

11. 나머지 설정에 대해서는 기본값을 선택한 후 다음 단계로 진행합니다.
12. 파트너 관계 마법사의 5 단계에서 서명 처리를 활성화하고 적절한 개인 키/인증서 쌍에 대한 별칭을 선택합니다.
13. "확인" 단계로 이동합니다. 구성을 검토하고 "마침"을 클릭하여 파트너 관계를 저장합니다.

    "파트너 관계" 목록으로 돌아갑니다.

14. "작업", "활성화"를 선택하여 파트너 관계를 활성화합니다.

페더레이션된 파트너 관계에 대해 STS 가 정의됩니다. 이제 CA SiteMinder for Secure Proxy Server 에서 STS 구성 요소를 구성합니다.

Office 365 와 STS 사이에서 트러스트 관계 구성(SOAP 기반 SSO)

SOAP 기반 클라이언트와 Office 365 사이에서 SSO 를 활성화하려면 Office 365 로그인 서비스와 STS 를 사용한 온-프레미스 서버 사이에 트러스트 관계를 구성하십시오. Office 365 구독을 구매하고 디렉터리 동기화를 구성한 후에 이 관계를 설정하십시오.

Windows Powershell 명령을 사용하여 트러스트 관계를 구성합니다. 최초에 트러스트 관계를 구성하는 명령은 Set-MsolDomainFederationSettings 입니다. 엔터프라이즈에서 이 명령을 실행하십시오. 올바른 절차에 대한 자세한 내용은 Windows Powershell 에 대한 Microsoft 설명서를 참조하십시오.

이 명령은 다음과 같은 명령 인수를 사용합니다.

- 도메인
- ActiveLogOnUri(온-프레미스 STS 의 ws-사용자 이름 끝점)
- PassiveLogOnUri
- IssuerUri
- MetadataExchangeUri
- SigningCertficate

이러한 명령 인수는 STS 를 사용한 Office 365 에서 온-프레미스 보안 프록시 서버 시스템으로의 트러스트 관계를 구성하는 데 충분합니다.

명령 인수에 대한 STS 끝점을 파악하려면 SiteMinder 관리 UI 에서 기존 WS-페더레이션 파트너 관계를 보십시오. 이러한 끝점은 WS-페더레이션 IP 에서 RP 로의 파트너 관계의 값에 기초합니다. STS 를 사용한 보안 프록시 서버 시스템을 트러스트하려면 Office 365 를 구성할 때 이러한 끝점을 사용하십시오.

참고: 특정 WS-페더레이션 파트너 관계의 구성을 보려면 "작업"을 선택한 다음 해당 파트너 관계 옆의 "보기"를 선택하십시오.

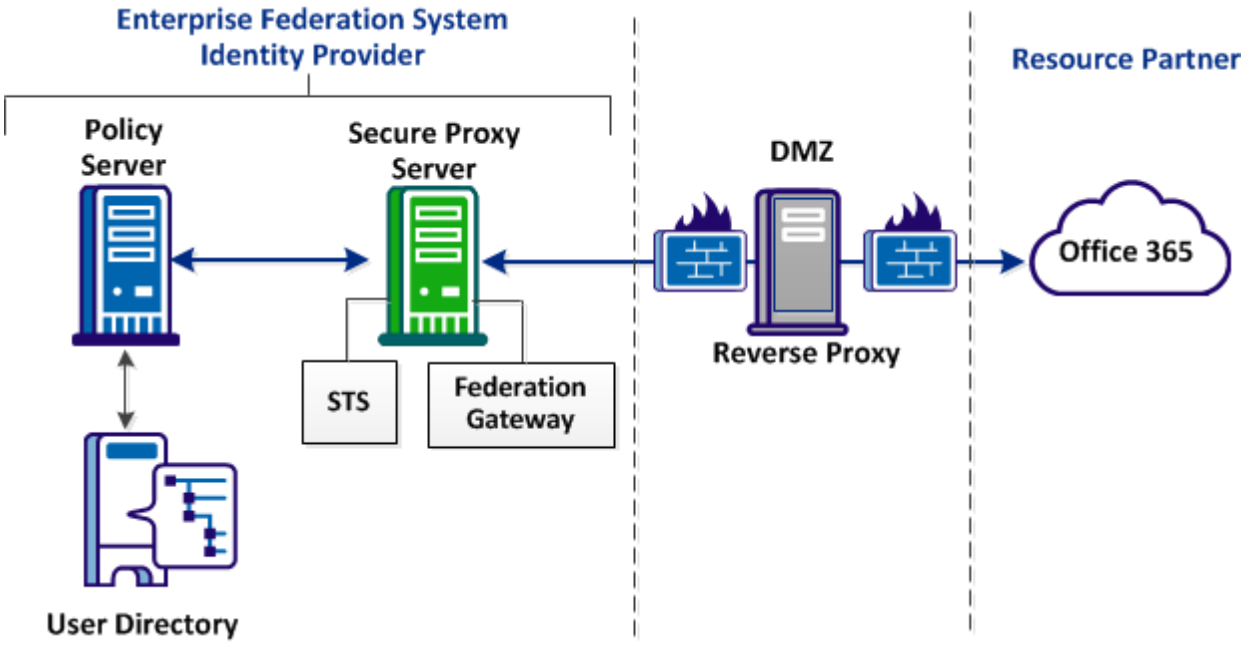
CA SiteMinder for Secure Proxy Server 구성

SiteMinder 가 WS-페더레이션 능동 프로필을 사용하여 싱글 사인온을 구현하려면 STS 웹 서비스가 필요합니다. Office 365 에서 요청을 관리하려면 CA SiteMinder for Secure Proxy Server 에서 STS 를 배포하십시오.

참고: STS 웹 서비스는 엔터프라이즈에서 CA SiteMinder for Secure Proxy Server 에 호스트되어야 합니다. 이 서비스는 다른 SiteMinder 플랫폼에서 호스트될 수 없습니다.

클라이언트 응용 프로그램이 Office 365 에 연결을 시도하면 Office 365 는 STS 로 토큰 요청을 발행합니다. STS 는 Office 365 가 소비할 수 있는 SAML 1.1 어설션이 있는 보안 토큰을 발행합니다. 클라이언트 응용 프로그램이 Office 365 에 액세스할 수 있습니다.

다음 그림은 이 페더레이션된 솔루션에 대해 권장되는 배포를 보여 줍니다. 트래픽을 STS 로 라우트하기 위해 여러 가지 방법을 사용할 수 있습니다.



다음 단계를 수행하십시오.

1. STS 설정 요구 사항을 확인합니다. (페이지 166)
2. JSafeJCE 보안 공급자를 사용하도록 JVM 을 구성합니다 (페이지 166).
3. STS 를 배포합니다 (페이지 167).

STS 설정 요구 사항을 확인합니다.

CA SiteMinder for Secure Proxy Server 에 STS 를 배포하기 전에 다음 요구 사항을 충족시키십시오.

- CA SiteMinder for Secure Proxy Server 를 설치 및 구성합니다.

  참고: 부하 부산 기능을 위해 두 개 이상의 보안 프록시 서버 시스템을 사용할 것을 권장하지만 필수 사항은 아닙니다.

- CA SiteMinder® Federation 시스템의 관리자로부터 파트너 관계 이름을 획득합니다. STS 를 배포할 때 STS 컨텍스트에 이 이름을 지정합니다.
- CA SiteMinder for Secure Proxy Server 에서 SSL 을 활성화합니다.
- Office 365 에서 오는 트래픽이 STS 를 사용한 온-프레미스 CA SiteMinder for Secure Proxy Server 에 도달하는지 확인합니다.
- STS 를 사용한 보안 프록시 서버 시스템은 내부 트래픽과 엔터프라이즈 방화벽 외부의 트래픽이 도달할 수 있어야 합니다. 외부 트래픽을 전달하기 위해 DMZ 에 프록시를 설치해야 할 수 있습니다. 방화벽 외부의 트래픽을 STS 로 전달할 수 있도록 프록시를 구성합니다.

JSafeJCE 보안 공급자를 사용하도록 JVM 구성

암호화를 활성화하려면 CA SiteMinder for Secure Proxy Server 를 실행 중인 JVM 이 JSafeJCE 보안 공급자를 사용하도록 구성하십시오.

다음 단계를 수행하십시오.

1. Oracle 웹 사이트에서 사용 중인 Java 버전에 대한 "JCE(Java Cryptography Extension) Unlimited Strength Jurisdiction Policy Files" 패키지를 다운로드합니다.
2. 다음 위치로 이동합니다.

   Windows

   JVM_HOME\lib\security

   UNIX

   JVM_HOME/lib/security

   JVM_HOME

   JRE(Java Runtime Environment)가 JDK 에서 설치된 위치를 정의합니다.

3. "JCE Unlimited Strength Jurisdiction Policy Files" 패키지의 파일을 사용하여 다음 파일을 패치하십시오.
   - local_policy.jar
   - US_export_policy.jar
4. java.security 파일을 엽니다.
5. "List of Providers" 섹션에 다음 줄을 추가하여 JSafeJCE 를 보조 보안 공급자로 추가합니다.

   ```
   security.provider.2=com.rsa.jsafe.provider.JsafeJCE
   ```
6. 다른 보안 공급자의 우선 순위를 한 수준 높입니다.
7. 기존 보안 공급자 목록의 끝에 다음 줄을 추가합니다. 이 줄은 JSafeJCE 의 초기 FIPS 모드를 설정합니다.

   ```
   com.rsa.cryptoj.fips140initialmode=NON_FIPS140_MODE
   ```
8. 변경 내용을 저장합니다.

다음 예는 JVM 을 구성한 후에 java.security 파일의 "List of Providers" 섹션을 보여 줍니다.

```
security.provider.1=sun.security.provider.Sun
security.provider.2=com.rsa.jsafe.provider.JsafeJCE
security.provider.3=sun.security.rsa.SunRsaSign
security.provider.4=com.sun.net.ssl.internal.ssl.Provider
security.provider.5=com.sun.crypto.provider.SunJCE
security.provider.6=sun.security.jgss.SunProvider
security.provider.7=com.sun.security.sasl.Provider
security.provider.8=org.jcp.xml.dsig.internal.dom.XMLDSigRI
security.provider.9=sun.security.smartcardio.SunPCSC
security.provider.10=sun.security.mscapi.SunMSCAPI
com.rsa.cryptoj.fips140initialmode=NON_FIPS140_MODE
```

STS 배포

WS-페더레이션 능동 요청자 프로필을 지원하려면 CA SiteMinder for Secure Proxy Server 에 STS 를 배포하십시오.

다음 단계를 수행하십시오.

1. SPS 관리 UI 를 엽니다.
2. "웹 서비스", "보안 토큰 서비스"로 이동합니다.
3. "추가"를 클릭합니다.

4. 다음 필드를 완성합니다.

   STS 이름

   STS 웹 서비스의 이름을 정의합니다. 관리 UI 에 정의된 파트너 관계 이름을 입력합니다.

   STS 컨텍스트

   STS 컨텍스트 경로를 정의합니다. 관리 UI 에 정의된 WS-페더레이션 파트너 관계의 이름을 지정합니다. */partnership_name* 구문을 사용하여 값을 입력합니다.

   예: /Office365Cloud

5. "확인", "저장"을 클릭합니다.
6. 시스템을 다시 시작합니다.
7. Office 365 로의 싱글 사인온을 테스트합니다.

Office 365 로의 SSO 테스트 및 문제 해결(능동 요청자 프로필)

SSO 테스트

Lync 또는 Outlook 에 로그인하여 WS-페더레이션 구성과 STS 배포를 확인하십시오.

다음 단계를 수행하십시오.

1. 엔터프라이즈의 시스템에 있는 Lync 또는 Outlook 에 로그인합니다.
2. 로그인되었고 마치 로컬에 설치된 것처럼 응용 프로그램을 사용할 수 있는지 확인합니다.

SSO 문제 해결

WS-페더레이션 파트너 관계 및 연결 문제의 경우 다음 조사 방법을 사용하십시오.

- WS-페더레이션 피동 요청자 프로필을 사용하여 Microsoft Online 과의 싱글 사인온을 확인합니다.

  웹 브라우저에서 http://portal.microsoftonline.com 또는 Microsoft Exchange Online 으로 이동합니다. 엔터프라이즈 자격 증명을 사용하여 로그인을 시도합니다. 브라우저에서 성공적으로 로그인할 수 있지만 엔터프라이즈 클라이언트에서는 로그인할 수 없는 경우 온-프레미스 STS 의 설정을 확인하십시오.

- Office 365 가 페더레이션 파트너로서 SiteMinder 에 대해 알고 있는 사항과 페더레이션 사용자에 대해 알고 있는 사항을 검토합니다. 파트너 관계와 사용자의 상태를 파악하려면 다음 Microsoft Powershell 명령을 수행하십시오.

  Get-MsolDomainFederationSettings

  도메인(즉, 엔터프라이즈)에 대해 Microsoft 가 알고 있는 정보를 표시합니다. 설정을 검토하고 정확한지 확인하십시오. 잘못된 정보는 페더레이션된 통신에 문제를 발생시킬 수 있습니다.

  Get-MsolUser

  Microsoft 가 특정 사용자에 대해 알고 있는 정보를 표시합니다. 사용자 설정을 검토하고 정확한지 확인합니다. 잘못된 정보는 페더레이션된 통신에 문제를 발생시킬 수 있습니다.

- Microsoft Remote Connectivity Analyzer 를 사용하여 엔터프라이즈와 Microsoft 사이의 연결 유효성을 검사합니다. 이 도구를 사용하면 Outlook, Lync, Office 365 와의 연결 문제를 파악할 수 있습니다. 이 도구는 https://www.testexchangeconnectivity.com/에서 찾을 수 있습니다.

STS 구성 요소에 대한 문제는 다음 로그 및 파일을 사용하십시오.

- STS 로그를 검토하여 STS 가 실행 중이고 인증 실패가 있는지 여부를 확인합니다.
  secure-proxy_install_dir/proxy-engine/logs/*partnership_name*.log 에 있는 로그를 확인합니다.

  STS 초기화가 완료되었음을 알리는 메시지를 확인합니다. 이 메시지는 STS 가 실행 중임을 나타냅니다.

- agent-log4j.xml 구성 파일에서 로그 설정을 구성합니다. 가장 자세한 정보가 *partnership_name*.log 에 기록되도록 모든 범주에 대한 로그 수준을 DEBUG 로 설정합니다. agent-log4j.xml 파일은 다음 디렉터리에 있습니다.

  secure-proxy_install_dir/proxy-engine/conf/sts-config/*partnership_name*/config/

  또한, 검사점 설정 <category name="com.ca.CheckPointLogger,"를 우선 순위 값 "INFO"로 설정하십시오. 이 설정은 인증 작업 및 어설션 작업에 대한 검사점 로그 메시지를 작성합니다. 검사점 로그 메시지는 STS 구성 요소의 작업을 반영하는 코드가 포함된 정보 제공 메시지입니다.

  페더레이션 추적 로깅 (페이지 295) 섹션에는 검사점 메시지가 설명되어 있습니다.

- WSDL 파일을 검토하고 온-프레미스 STS 가 응답하는지 확인합니다. 브라우저를 열고 http://*sts.company.com*/*partnership_name*?wsdl 로 이동합니다. 문자열 *sts.company.com* 은 STS URL 에 대한 자리 표시자입니다. STS URL 은 관리 UI 에서 구성된 WS-페더레이션 파트너 관계에서 찾을 수 있습니다.

SAML 2.0 HTTP-POST 바인딩 구성

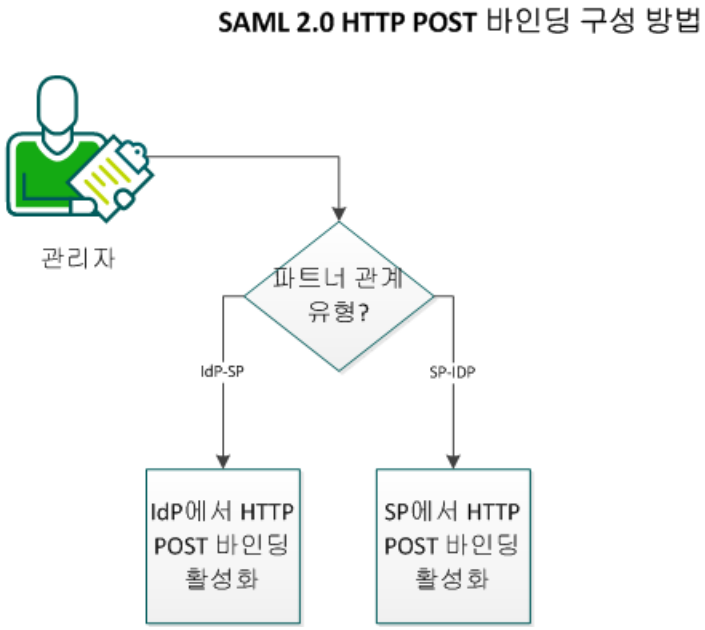
싱글 사인온 및 싱글 로그아웃 요청에 대해 요청 및 응답 교환 방법으로 SAML 2.0 HTTP-POST 바인딩을 사용할 수 있습니다. 이 바인딩은 SAML 프로토콜을 표준 메시징 형식 및 통신 프로토콜로 매핑합니다.

참고: 인증 요청 바인딩은 SSO 바인딩과 다릅니다. SSO 바인딩은 특정 사용 방식을 처리하기 위해 어설션, 프로토콜, 바인딩이 상호 작용하는 방법을 지정하는 프로필을 결정합니다.

이 절차를 수행하려면 페더레이션 환경에 대해 잘 알고 있어야 하며, 다음과 같은 파트너 관계 중 하나 이상을 생성되어 활성화되어 있어야 합니다.

- IdP-SP
- SP-IdP

다음 그림에서는 SAML 2.0 HTTP POST 바인딩을 사용하도록 설정하는 방법을 설명합니다.



다음 단계를 수행하십시오.

1. 파트너 관계 유형에 적합한 태스크를 수행하십시오.
   - IdP 에서 HTTP POST 바인딩을 활성화합니다 (페이지 172).
   - SP 에서 HTTP POST 바인딩을 활성화합니다 (페이지 173).

IdP 에서 HTTP POST 바인딩 활성화

IdP 에서 HTTP-POST 바인딩을 활성화할 수 있습니다.

중요! 인증 요청 바인딩을 구성하기 전에 세션 저장소를 활성화하십시오. IdP 가 HTTP-POST 바인딩을 사용하여 전달된 인증 요청을 처리할 수 있기 위해서는 IdP 가 세션 저장소에 요청을 저장해야 합니다.

세션 저장소가 사용되도록 설정

다음 단계를 수행하십시오.

1. 정책 서버 관리 콘솔을 열고 "데이터" 탭을 선택합니다.
2. 다음 필드를 설정합니다.

   데이터베이스

   세션 저장소

   저장소

   저장소 리포지토리를 선택합니다.

   세션 저장소 사용

   이 확인란을 선택하십시오.
3. 데이터 원본 정보를 입력합니다.
4. "확인"을 클릭하여 변경 내용을 저장합니다.

관리 UI 에서 바인딩을 구성합니다.

다음 단계를 수행하십시오.

1. 관리 UI 를 엽니다.
2. 수정할 파트너 관계가 활성화되어 있는 경우 비활성화합니다.
3. "수정"을 클릭하여 파트너 관계 마법사를 엽니다.
4. "SSO 및 SLO" 단계로 이동합니다.
5. SSO 섹션에서 인증 요청 바인딩에 대해 "HTTP-POST"를 선택합니다.

   참고: 인증 요청에 대해 HTTP-리디렉션 및 HTTP-POST 바인딩을 모두 선택할 수 있습니다.
6. (선택 사항) SLO 섹션에서 "HTTP-POST" 확인란을 선택합니다.

   참고: 여러 SLO 바인딩을 선택할 수 있습니다.

7. SLO 바인딩과 일치하는 바인딩을 사용하여 SLO 서비스 URL 을 지정합니다. HTTP-리디렉션 및 HTTP-POST 바인딩을 선택한 경우 각 SLO 바인딩마다 하나씩 두 개의 SLO 서비스를 만드십시오.
8. 필요한 경우 다른 파트너 관계 정보를 입력합니다.
9. 확인 단계에서 "마침"을 클릭합니다.

SSO HTTP-POST 바인딩이 이제 활성화되었습니다.

SP 에서 HTTP POST 바인딩 활성화

SP 에서 인증 및 SLO 요청에 대한 HTTP-POST 바인딩을 활성화할 수 있습니다.

다음 단계를 수행하십시오.

1. 관리 UI 를 엽니다.
2. 수정할 파트너 관계가 활성화되어 있는 경우 비활성화합니다.
3. "수정"을 클릭하여 파트너 관계 마법사를 엽니다.
4. 파트너 관계 마법사의 "SSO 및 SLO" 탭으로 이동합니다.
5. SSO 섹션에서 인증 요청 바인딩에 대해 "HTTP-POST"를 선택합니다.

   참고: 인증 요청에 대해 HTTP-리디렉션 및 HTTP-POST 바인딩을 모두 선택할 수 있습니다.
6. 인증 요청 바인딩과 일치하는 바인딩을 사용하여 원격 SLO 서비스 URL 을 지정합니다. 예를 들어, HTTP-리디렉션 및 HTTP-POST 바인딩을 선택한 경우 각 바인딩마다 하나씩 두 개의 SLO 서비스 URL 을 만드십시오.
7. (선택 사항) SLO 섹션에서 "HTTP-POST" 확인란을 선택합니다.

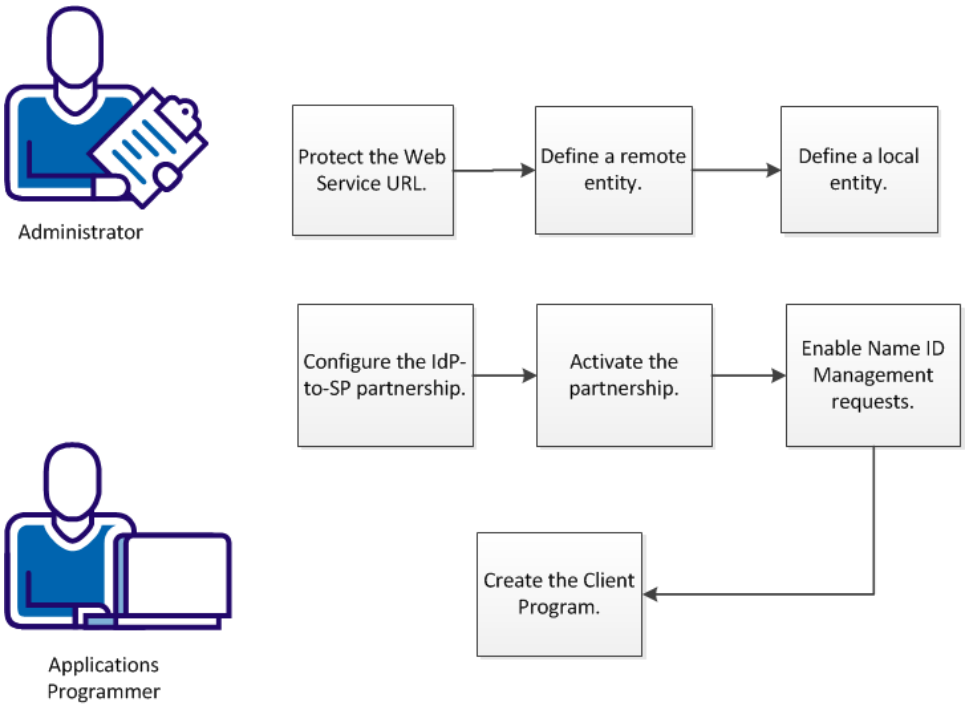
   참고: 여러 SLO 바인딩을 선택할 수 있습니다.
8. SLO 바인딩과 일치하는 바인딩을 사용하여 SLO 서비스 URL 을 지정합니다. 예를 들어, HTTP-리디렉션 및 HTTP-POST SLO 바인딩을 선택한 경우 각 바인딩마다 하나씩 두 개의 SLO 서비스 URL 을 만드십시오.
9. 필요한 경우 다른 파트너 관계 정보를 입력합니다.
10. 확인 단계에서 "마침"을 클릭합니다.

SSO HTTP-POST 바인딩이 활성화되었습니다.

SAML 2.0 이름 ID 관리 프로필 구성

SAML 2.0 이름 식별자 프로필을 사용하면 페더레이션된 파트너 관계에서 개별 사용자의 프로비저닝을 취소할 수 있습니다. 다양한 이유로 파트너 관계에서 사용자를 제거하려 할 수 있습니다. 예를 들어, 직원이 퇴사했거나 서비스 공급자 기반의 SSO 기능을 더 이상 필요로 하지 않을 수 있습니다. 프로비저닝 취소 요청은 클라이언트 응용 프로그램을 통해 수행합니다.

다음 다이어그램에서는 SAML 2.0 이름 식별자 프로필을 구현하는 프로세스를 보여 줍니다.



이름 ID 관리 프로필을 사용하여 사용자의 프로비저닝을 취소하려면 다음 단계를 수행해야 합니다.

1. 이름 식별자 관리를 위한 관리 웹 서비스 URL 을 보호합니다. (페이지 175)
2. 이름 ID 관리에 대한 원격 엔터티를 구성합니다. (페이지 175)
3. 로컬 엔터티를 생성합니다. (페이지 176)
4. 이름 ID 관리에 대한 파트너 관계를 구성합니다. (페이지 176)
5. 파트너 관계를 활성화합니다. (페이지 177)
6. 이름 ID 관리 요청을 활성화합니다. (페이지 178)
7. 이름 식별자 웹 서비스와 상호 작용하는 클라이언트 응용 프로그램을 생성합니다. (페이지 178)

이름 식별자 관리를 위한 관리 웹 서비스 URL 보호

고객 응용 프로그램은 이름 식별자 관리를 위한 관리 웹 서비스를 사용하여 파트너 관계에서 사용자의 프로비저닝 취소를 요청할 수 있습니다. 이 웹 서비스는 REST 인터페이스를 구현합니다.

이 서비스의 URL 은 /affwebservices/saml2nidws 입니다. 이 URL 을 SiteMinder 기본 자격 증명을 사용하여 보호합니다. 이 서비스의 모든 사용자를 도메인과 연결된 사용자 디렉터리에 포함하십시오. SiteMinder 정책 관리자는 기본적으로 포함되지 않습니다. 연결된 디렉터리에 수동으로 추가할 수 있습니다.

이름 ID 관리에 대한 원격 엔터티 구성

이름 ID 관리를 지원하는 파트너 관계를 생성하는 첫 번째 단계는 원격 및 로컬 파트너나 엔터티를 정의하는 것입니다. 엔터티를 수동으로 구성하거나 XML 메타데이터를 가져올 수 있습니다. 수동 구성 단계는 다음과 같습니다.

다음 단계를 수행하십시오.

1. 관리 UI 에서 "페더레이션", "파트너 관계 페더레이션", "엔터티"로 이동합니다.
2. "엔터티 만들기"를 클릭합니다.
3. "원격"(구현에 따라 IdP 또는 SP)을 선택합니다.

4. "다음"을 클릭하여 엔터티의 세부 사항을 구성합니다.
5. "엔터티 ID" 및 "엔터티 이름"(필수)의 값을 입력합니다.
6. "이름 ID 관리 서비스 URL"에 대한 행에서 "행 추가"를 클릭합니다.
7. SOAP 바인딩을 선택합니다. 원격 엔터티가 다른 바인딩을 지정할 수 있지만, 이 경우 가져오기는 하지만 사용되지 않습니다.
8. 이름 ID 관리 서비스의 URL 을 지정하는 "위치 URL"을 입력합니다. 이 값은 다음과 같습니다.

   http://*sp_server:port*/affwebservices/public/saml2nidsoap
9. "응답 위치 URL" 필드를 비워 둡니다. SOAP 바인딩에 대한 응답 위치 URL 과 위치 URL 은 동일합니다.
10. 이 목록에서 지원되는 이름 ID 형식을 선택합니다.
11. 구현에 필요한 다른 필드를 모두 완료합니다.
12. "다음"을 클릭하여 엔터티 구성을 확인합니다.
13. "마침"을 클릭합니다.

로컬 엔터티 만들기

이름 ID 관리를 지원하는 파트너 관계를 생성하는 첫 번째 단계는 원격 및 로컬 파트너나 엔터티를 정의하는 것입니다. 엔터티를 수동으로 구성하거나 XML 메타데이터를 가져올 수 있습니다. 파트너 관계에서 엔터티를 생성하는 데 익숙하지 않은 경우 자세한 내용은 페더레이션 엔터티 구성 (페이지 69)을 참조하십시오.

중요! 사용자 프로비저닝 취소 또는 연결 취소를 위한 임의의 이름 ID 형식을 선택할 수 있습니다. 동적 계정 연결은 "영구 식별자" 형식만 지원합니다. 계정 연결 및 연결 취소를 구현하는 경우 "영구 식별자 이름 ID" 형식을 선택하십시오.

이름 ID 관리에 대한 파트너 관계 구성

이름 ID 관리 기능을 사용하려면 새 파트너 관계나 기존 파트너 관계에 대한 구성이 필요합니다. 로컬 또는 원격 엔터티가 파트너 관계에서 사용자의 프로비저닝을 취소하는 요청을 시작할 수 있습니다.

다음 단계를 수행하십시오.

1. "SSO 및 SLO" 대화 상자로 이동합니다.
2. 아직 구성하지 않은 경우 "인증" 및 "SSO" 섹션을 구성합니다.
3. "이름 ID 관리" 섹션으로 이동합니다.
4. "MNI" 필드에서 "SOAP"를 선택합니다.

   이 항목을 선택하면 파트너 관계에서 이름 ID 관리가 사용됩니다. 이러한 옵션에 대한 설명은 온라인 도움말을 참조하십시오.
5. (필수) SOAP 시간 만료 값을 지정합니다. 이 값은 런타임에서 원격 공급자에 대한 요청 시간이 만료될 때까지 대기하는 시간(초)입니다.

   기본값: 60 초
6. (필수 사항) "다시 시도 횟수"를 지정합니다. 이 값은 실패로 간주되기 전까지 백그라운드 요청이 시도되는 횟수입니다. 기본값은 3 입니다.
7. (필수 사항) "다시 시도 경계"를 지정합니다. 이 값은 재시도 간격(분)입니다. 기본값은 15 분입니다.
8. "알림 사용" 옵션을 선택한 경우 "알림 URL"을 지정합니다. 이 URL 은 고객 응용 프로그램에 HTTP 알림을 보내는 위치입니다. 이 알림에는 프로비저닝 취소 요청이 완료된 후 다음과 같은 요청의 상태가 포함됩니다.
   - 상태 1 은 프로비저닝 취소가 성공했음을 나타냅니다.
   - 상태 0 은 프로비저닝 취소가 실패했음을 나타냅니다.
9. "알림 시간 만료"를 지정합니다. 이 값은 요청 시간이 만료된 것으로 간주될 때까지 경과하는 시간(초)입니다.

   기본값: 60 초
10. "알림 인증 유형"("인증 없음" 또는 "기본")을 지정합니다. "기본"을 선택하는 경우 사용자 이름과 암호를 제공하십시오.

참고: 기능이 제대로 작동할 수 있도록 MNI 섹션에서 "이름 ID 삭제" 옵션이나 "알림 사용" 옵션 또는 두 옵션을 모두 선택하십시오.

이러한 단계는 "이름 ID 관리" 구성을 완료합니다.

파트너 관계 활성화

자세한 내용은 파트너 관계 활성화 (페이지 86)를 참조하십시오.

이름 ID 관리 요청 활성화

"비동기 요청 처리기"라고 하는 웹 에이전트 옵션 팩 내부 구성 요소는 이름 ID 관리 서비스에 대한 모든 요청을 처리합니다. 단 하나의 웹 에이전트 옵션 팩만 이 서비스를 한 번에 하나만 실행할 수 있습니다. 관리 UI 의 설정에 추가하여, 다음 위치의 AffWebServices.properties 파일에서 설정을 지정하여 이름 ID 관리 처리를 활성화합니다.

- SPS: <SECURE_PROXY_HOME>/Tomcat/webapps/affwebservices/WEB-INF/classes
- WA+WAOP: <WEB_AGENT_HOME>/affwebservices/WEB-INF/classes

AffWebServices.properties 파일은 이름 ID 관리와 관련된 다음 설정을 포함하고 있습니다.

ProcessBackgroundNameIDOperations

이 시스템이 이름 ID 작업을 처리하는지 여부를 지정합니다.

기본값: False

중요! 옵션 팩 또는 SPS 에 대한 이름 ID 관리를 사용하려면 이 설정을 True 로 설정해야 합니다.

BackgroundProcessingInterval

이름 ID 요청에 대한 비동기 프로세서 검사 간 간격(초)을 지정합니다. 이 값은 수정할 수 있습니다.

기본값: 60 초

옵션 팩 또는 SPS 를 업그레이드하는 경우 설치 관리자는 이러한 설정을 기본값을 사용하여 새 속성 파일에 추가합니다.

이름 식별자 웹 서비스와 상호 작용하는 클라이언트 응용 프로그램 만들기

클라이언트 응용 프로그램의 콘텐츠는 구현에 따라 달라집니다. 사용자 제거를 요청하려면 이름 식별자 관리를 위한 관리 웹 서비스를 사용하십시오. 웹 서비스는 다음 두 가지 HTTP 메서드를 구현합니다.

- POST - 프로비저닝 취소 요청을 시작합니다.
- GET - 요청의 상태를 폴링합니다.

메서드는 OData 프로토콜을 준수합니다. 이러한 메서드에 대한 자세한 내용은 다음과 같습니다.

페더레이션 구성원 자격 종료

관리자는 다음 URL 을 사용하여 사용자의 페더레이션 구성원 자격을 종료할 수 있습니다.

```
POST http://<server+port>/affwebservices/saml2nidws/terminate
```

이 비동기 요청은 XPS 에서 ManageNameID 이벤트를 생성합니다.

POST 본문에는 다음 값이 포함됩니다.

UserDN

SMSession 이 없으므로 사용자를 명확히 구분합니다. LDAP 에 대한 DN 예: uid=user0001,ou=Engineering,o=security.com

OperationType

특정 사용 사례를 나타냅니다. 유효한 값은 다음과 같습니다.

- sp - 특정 서비스 공급자와의 페더레이션을 종료하려는 idp 임을 나타냅니다.
- idp - 특정 아이덴티티 공급자와의 페더레이션을 종료하려는 서비스 공급자임을 나타냅니다.

ProviderID

작업에 참여하는 공급자를 결정합니다. sp 및 idp 값이 사용될 경우 ProviderID 는 원격 공급자를 나타냅니다. OperationType 이 'sp'이면 ProviderID 는 원격 서비스 공급자 개체를 나타냅니다. OperationType 이 'idp'이면 ProviderID 는 원격 아이덴티티 공급자 개체를 나타냅니다.

요청의 POST 본문에 있는 정보는 JSON 또는 AtomPub 형식입니다. 다음은 JSON 형식의 예입니다.

```
{
"UserDN":"uid=user0001,ou=Engineering,o=security.com",
"OperationType":"sp",
"ProviderID":"http://company.example.com/SPID"
}
```

이 요청은 다음과 같은 지속되는 개체를 나타내는 리소스를 반환합니다.

```
http://<server+port>/affwebservices/saml2nidws/terminate(<XID>)
```

<XID>는 생성된 개체의 XPS XID 입니다. 클라이언트는 이 URL 을 사용하여 이 개체의 변경 내용을 폴링할 수 있습니다.

이 요청은 다음과 같은 완전한 AtomPub 형식일 수도 있습니다.

```
<?xml version="1.0" encoding="utf-8"?>
<entry xmlns="http://www.w3.org/2005/Atom"
xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices"
xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata"
>
<title type="text"></title><author><name></name></author>
<category term="NameidProducer.terminate"
scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"><
/category>
<content type="application/xml">
<m:properties>
<d:UserDN>uid=user0001,ou=Engineering,o=security.com</d:UserDN><d:Provi
derID>http://company.example.com/SPID</d:ProviderID>
<d:OperationType>SP</d:OperationType>
</m:properties>
</content>
</entry>
```

POST 서비스는 다음과 같은 HTTP 반환 코드를 설정합니다.

HTTP 상태	설명
201	리소스가 생성됨
400	잘못된 요청
415	지원되지 않는 미디어 유형
500	내부 서버 오류

상태 폴링

관리자는 이 서비스를 통해 다음 URL 을 사용하여 비동기 요청의 상태를 요청할 수 있습니다.

```
GET http://<server+port>/affwebservices/saml2nidws/terminate(<XID>)
```

리소스 상태를 폴링하는 데 사용되는 URL 입니다.

응답은 PENDING, COMPLETED, FAILED 중에서 요청의 상태를 반환합니다.

중요! 이 요청을 하기 전에 에이전트 구성 개체의 CssChecking 매개 변수가 NO 로 설정되었는지 확인하십시오. 이 설정은 OData 사이의 잠재적인 충돌 및 사이트 간 스크립트 공격을 방지합니다.

GET 서비스는 다음과 같은 HTTP 반환 코드를 설정합니다.

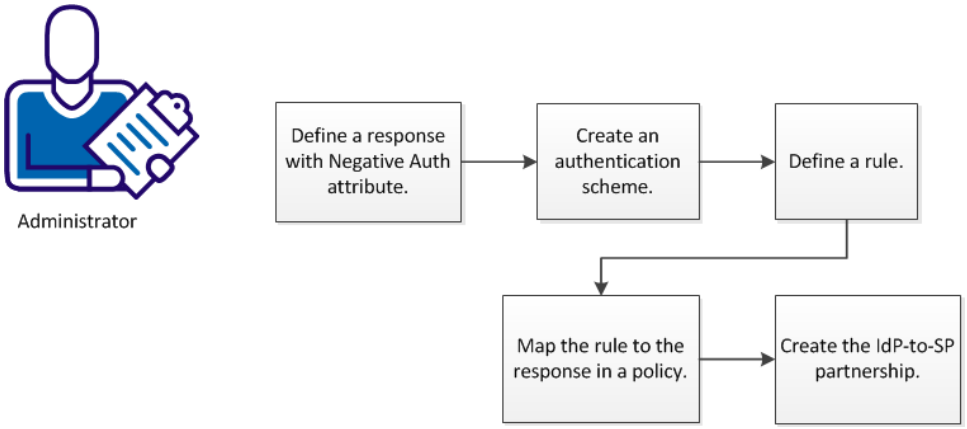
HTTP 상태	설명
200	확인
400	잘못된 요청
403	금지됨(웹 에이전트에 대해 CSS 검사가 설정된 경우)
415	지원되지 않는 미디어 유형
500	내부 서버 오류

인증 실패에 대한 SAML 2.0 응답 구성

다음 프로세스를 사용하여 인증 실패 시 서비스 공급자에 대한 어설션 이외의 응답을 구성할 수 있습니다. SAML 2.0 인증 요청에 성공하면 서비스 공급자에 대한 응답이 인증 어설션과 함께 전달됩니다. 이전에는 인증 요청이 거부된 경우 최종 사용자에게만 오류 메시지가 표시되었습니다. 서비스 공급자에게는 실패한 상태에 대한 알림이 전달되지 않았습니다. 제어권이 서비스 공급자에게 반환되었으므로 서비스 공급자는 사용자를 리디렉션할지 또는 다른 적절한 조치를 취할지 결정할 수 있습니다.

중요! 이 기능이 작동하려면 정책 서버, 웹 에이전트 및 웹 에이전트 옵션 팩이 모두 SM r12.52 이상에 있어야 합니다.

다음 다이어그램에서는 이 기능을 구성하는 데 필요한 단계를 보여 줍니다.



인증 실패 시 서비스 공급자에 대한 응답을 구성하는 프로세스에는 다음 절차가 포함됩니다.

1. 부정적 인증 응답 특성을 지정하는 응답을 정의합니다. (페이지 182)
2. 기본 또는 양식 인증 체계를 생성합니다 (페이지 183).
3. OnAuthReject 작업을 지정하는 규칙을 정의합니다 (페이지 184).
4. 이 규칙을 정책에서 이전에 정의한 응답에 매핑합니다 (페이지 185).
5. 부정적 인증 응답을 사용하도록 IdP-SP 파트너 관계를 구성합니다 (페이지 185).

부정적 인증 응답 특성을 지정하는 응답 정의

먼저 WebAgent-OnReject-eGovNegResponse 특성 유형을 사용하여 응답을 정의합니다. 응답 정의에서는 도메인이 정의되어 있다고 가정합니다.

다음 단계를 수행하십시오.

1. "정책", "도메인", "응답"으로 이동합니다.
2. "응답 만들기"를 클릭합니다.
3. 적절한 도메인을 선택하거나 도메인을 새로 만듭니다.
4. "다음"을 클릭합니다.
5. "일반" 섹션에 이 응답의 이름과 설명(선택 사항)을 입력합니다.
6. 적절한 에이전트 유형(일반적으로 SiteMinder 웹 에이전트)을 선택합니다.

7. "특성 목록" 섹션에서 "응답 특성 만들기"를 클릭합니다.
8. "특성 유형" 섹션의 드롭다운 목록에서 WebAgent-OnReject-eGovNegResponse 를 선택합니다.
9. "특성 필드" 섹션에서 "상대 대상 사용"을 선택하거나 웹 서버 이름을 입력합니다.
10. (선택 사항) "SSL 연결 사용"을 선택합니다.

    참고: 이 섹션에서 선택하는 사항은 "고급" 섹션의 창에 표시되는 스크립트의 기반이 됩니다. 자세한 내용은 온라인 도움말을 참조하십시오.
11. "특성 캐싱" 섹션에서 "Cache Value Recalculate Value"(캐시 값 값 재계산)를 선택합니다.
12. "확인"을 클릭하여 "응답 만들기: 응답 정의" 대화 상자로 돌아갑니다.
13. "마침"을 클릭합니다.

인증 실패 시 SP 에 대한 응답을 생성하도록 적절한 특성으로 응답을 정의했습니다.

기본 또는 양식 인증 체계 구성

인증 실패 시 SP 에 대한 응답을 생성하도록 기본 체계 또는 양식 체계를 구성할 수 있습니다.

다음 단계를 수행하십시오.

1. "인프라", "인증"을 차례로 클릭합니다.
2. "인증 체계"를 클릭합니다.

3. "인증 체계 만들기"를 클릭합니다.

   "인증 체계 유형의 새 개체 만들기"가 선택되어 있는지 확인합니다.

4. "확인"을 클릭합니다.
5. 이름 및 보호 수준을 입력합니다.
6. "인증 체계 유형" 목록에서 "Basic or Forms Template"(기본 또는 양식 템플릿)을 선택합니다.
7. 제출을 클릭합니다.

   인증 체계가 저장되고 이제 영역에 할당할 수 있습니다.

인증 이벤트 작업을 위한 규칙 구성

사용자가 리소스에 액세스하려고 할 때 수행되는 작업을 제어하는 규칙을 구성할 수 있습니다. 인증 실패에 대한 완전한 SAML 2.0 응답의 경우 OnAuthReject 작업을 선택하십시오.

영역이 인증 이벤트를 처리할 수 있어야 합니다. "인증 이벤트 처리" 옵션이 선택되어 있는지 확인합니다. 영역을 생성하는 방법에 대한 자세한 내용은 다음에 나오는 항목을 참조하십시오.

다음 단계를 수행하십시오.

1. "정책", "도메인", "규칙"을 클릭합니다.
2. "규칙 만들기"를 클릭합니다.
3. 목록에서 도메인을 선택한 후 "다음"을 클릭합니다.
4. 규칙으로 보호할 리소스가 포함된 영역을 선택하고 "다음"을 클릭합니다.

   참고: 보호할 리소스에 대한 영역이 없는 경우에는 해당 리소스를 보호하기 위한 규칙을 생성할 수 없습니다.

5. 규칙의 이름과 설명을 입력합니다.

   참고: "도움말"을 클릭하면 해당되는 각 요구 사항과 제한을 포함하여 설정과 컨트롤에 대한 설명을 볼 수 있습니다.

6. 인증 이벤트를 선택합니다.

   "작업 목록"이 인증 이벤트로 채워집니다.

   참고: 인증 이벤트는 영역 전체에 적용되므로 "리소스" 필드는 비활성화됩니다. "액세스 허용" 및 "액세스 거부" 옵션 역시 인증 이벤트에 적용되지 않으므로 비활성화됩니다.

7. OnAuthReject 작업을 선택합니다.
8. (선택 사항) "고급" 섹션에서 시간 제한 및/또는 활성 규칙 설정을 지정합니다.
9. "마침"을 클릭합니다.

   규칙이 저장되고 지정된 영역 및 리소스에 적용됩니다.

OnAuthReject 작업을 사용하여 규칙을 적절한 응답에 매핑

OnAuthReject 작업을 사용하여 생성한 규칙을 정책의 eGovNegResponse 특성에 연결하십시오.

다음 단계를 수행하십시오.

1. "정책", "도메인 정책"으로 이동합니다.
2. 정책을 선택합니다.
3. "규칙"으로 이동합니다.
4. OnAuthReject 작업으로 생성한 규칙이 규칙 목록에 있는지 확인합니다.
5. 해당 규칙 옆의 "응답 추가"를 클릭합니다.
6. eGovNegResponse 특성 유형으로 지정한 응답을 선택합니다.
7. 저장하고 종료합니다.

규칙을 적절한 응답과 연결했습니다.

부정적 인증 응답을 지원하도록 IdP-SP 파트너 관계 구성

IdP-SP 파트너 관계 구성의 SSO 구성 단계에서 부정적 인증 응답이 사용되도록 설정합니다. "부정적 인증 응답 사용" 확인란을 선택합니다.

자세한 내용은 싱글 사인온 구성 (페이지 123)을 참조하십시오.

제 12 장: 소셜 사인온 구성

사용자가 페더레이션 시스템 자격 증명이 아닌 자신의 소셜 네트워킹 자격 증명을 사용하여 페더레이션된 리소스에 사인온할 수 있도록 CA SiteMinder?Federation(페더레이션 시스템)을 구성할 수 있습니다.

소셜 사인온 기능은 다음과 같은 기능으로 구성됩니다.

- 사용자가 자신의 OAuth 권한 부여 서버 자격 증명을 사용하여 페더레이션된 리소스에 사인온할 수 있도록 Facebook 같은 OAuth 권한 부여 서버를 사용한 사용자의 인증
- 사용자가 SAML 2.0 또는 Facebook 같은 여러 아이덴티티 공급자를 선택할 수 있는 자격 증명 선택기 페이지의 구성 사용자는 인증을 위한 아이덴티티 공급자를 선택하여 페더레이션된 리소스에 사인온할 수 있습니다.

이 기능은 서로 독립적이며 기능 중 하나 또는 모두를 구현하도록 페더레이션 시스템을 구성할 수 있습니다.

OAuth 권한 부여 서버를 사용한 사용자 인증

OAuth 권한 부여 서버를 사용하여 사용자를 인증하려면 페더레이션 시스템과 OAuth 권한 부여 서버 사이에 싱글 사인온을 구성하십시오.

페더레이션 시스템은 다음과 같은 OAuth 권한 부여 서버를 기본적으로 지원합니다.

OAuth 1.0a

- Twitter

OAuth 2.0

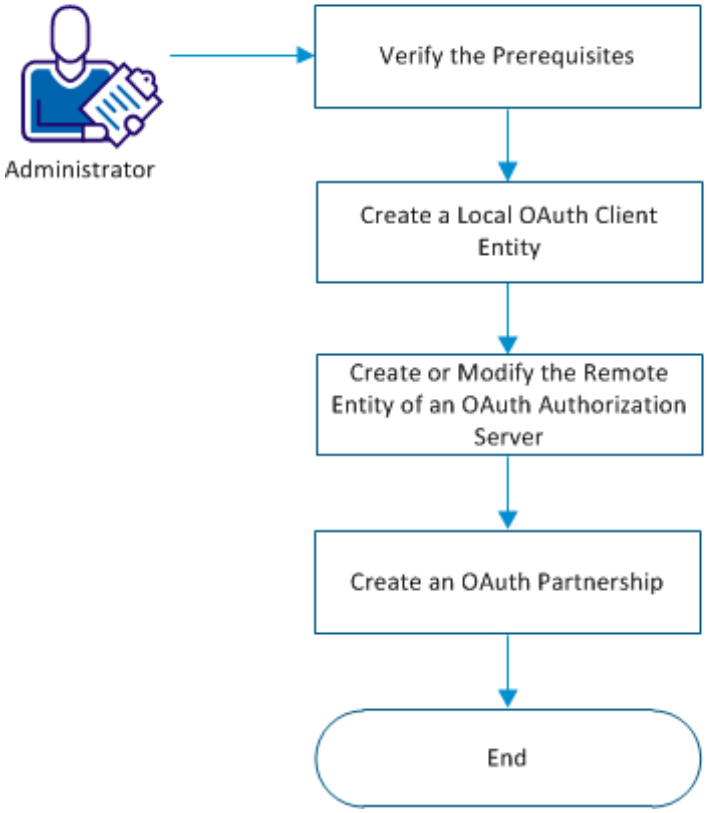
- Facebook
- Google
- LinkedIn
- Windows Live

다음 프로세스는 페더레이션 시스템이 페더레이션된 리소스에 대한 사용자 액세스 요청을 처리하는 방법을 설명합니다.

1. 페더레이션 시스템은 사용자 요청을 사용자 요청에 지정된 OAuth 권한 부여 서버에 리디렉션합니다.
2. OAuth 권한 부여 서버는 사용자를 인증하고 사용자에 대한 클레임을 포함한 인증 응답을 페더레이션 시스템에 전달합니다.
3. 페더레이션 시스템은 인증 응답을 확인하고, 인증 프로세스를 완료하고, 페더레이션된 시스템에 대한 사용자 액세스를 허가합니다.

다음 순서도는 OAuth 권한 부여 서버를 사용하여 사용자를 인증하는 방법을 설명합니다.

Authenticate Users Using an OAuth Authorization Server



다음 단계를 수행하십시오.

1. 사전 요구 사항을 확인합니다 (페이지 189).
2. 로컬 OAuth 클라이언트 엔터티를 만듭니다 (페이지 190).
3. (선택 사항) OAuth 권한 부여 서버의 원격 엔터티를 생성 또는 수정합니다 (페이지 190).
4. 싱글 사인온에 대한 OAuth 파트너 관계를 만듭니다 (페이지 192).

사전 요구 사항 확인

페더레이션 시스템과 OAuth 권한 부여 서버 사이에 싱글 사인온을 구성하기 위해 파트너 관계를 구성하기 전에 다음 단계를 수행하십시오.

- 페더레이션 시스템에서 SSL 을 활성화하십시오.
- 페더레이션 시스템이 기본적으로 지원하는 OAuth 권한 부여 서버를 사용하려면 파트너 관계를 호출하기 전에 다음 단계를 수행하십시오.
  - 독립 실행형 배포의 경우 OAuth 권한 부여 서버의 기본 CA 인증서를 가져왔는지 확인하십시오.
  - 통합 배포의 경우 smkeytool 을 사용하여 OAuth 권한 부여 서버의 기본 CA 인증서를 가져오십시오.
- 페더레이션 시스템이 기본적으로 지원하지 않는 OAuth 권한 부여 서버를 사용하려면 파트너 관계를 시작하기 전에 OAuth 권한 부여 서버의 SSL CA 인증서를 획득하고 가져오십시오.

로컬 OAuth 클라이언트 엔터티 만들기

페더레이션 시스템과 OAuth 권한 부여 서버 사이에 파트너 관계에 대한 로컬 OAuth 클라이언트 엔터티를 만듭니다.

다음 단계를 수행하십시오.

1. "페더레이션", "엔터티"로 이동하여 "엔터티 만들기"를 클릭합니다.
2. "엔터티 위치"에서 "로컬"을 선택합니다.
3. "새 엔터티 유형"에서 "OAuth 클라이언트"를 선택합니다.
4. OAuth 버전을 선택하고 "다음"을 클릭합니다.
5. 필수 값을 입력하고 "다음"을 클릭합니다.
6. 입력한 값을 확인하고 "마침"을 클릭합니다.

   리디렉션 URL 이 구성됩니다. 이 URL 을 사용하여 OAuth 트랜잭션을 시작하십시오.

권한 부여 서버의 원격 엔터티 생성 또는 수정

시스템은 기본적으로 제공되는 다음과 같은 OAuth 권한 부여 서버 각각에 대한 원격 엔터티를 제공합니다.

OAuth 1.0a

- Twitter

OAuth 2.0

- Facebook
- Google
- LinkedIn
- Windows Live

각 원격 엔터티의 값은 엔터티의 알려진 값을 사용하여 미리 구성되어 있습니다. 사용하는 페더레이션 환경에 맞게 이 값을 수정하거나 OAuth 권한 부여 서버에 대한 원격 엔터티를 만들 수 있습니다.

다음 단계를 수행하십시오.

1. 다음 작업 중 *하나*를 수행합니다.

   새 원격 엔터티를 만듭니다.

   a. "페더레이션", "엔터티", "엔터티 만들기"로 이동합니다.

   b. "엔터티 위치"로 "원격"을 선택한 다음 "새 엔터티 유형"으로 "OAuth 권한 부여 서버"를 선택합니다.

   c. "다음"을 클릭합니다.

   d. 값을 입력하고 "다음"을 클릭합니다.

   원격 엔터티의 미리 채워진 값을 수정합니다.

   a. "페더레이션", "엔터티"로 이동하여 수정할 엔터티를 검색합니다.

   b. 엔터티의 "작업" 옵션을 클릭한 다음 "수정"을 클릭합니다.

   c. "다음"을 클릭하여 "엔터티 구성" 탭으로 이동합니다.

   d. 값을 수정하고 "다음"을 클릭합니다.

2. 변경 내용을 확인하고 "마침"을 클릭합니다.

싱글 사인온에 대한 OAuth 파트너 관계 만들기

페더레이션 시스템이 권한 부여 서버로부터 사용자 정보를 가져올 수 있도록 하려면 OAuth 권한 부여 서버(어설션 당사자)와 페더레이션 시스템(신뢰 당사자) 사이에 OAuth 파트너 관계를 만드십시오.

다음 단계를 수행하십시오.

1. "페더레이션", "파트너 관계"로 이동한 다음 "파트너 관계 만들기"를 클릭합니다.
2. "OAuth 클라이언트 - 권한 부여 서버" 파트너 관계 유형을 선택합니다.
3. 파트너 관계 정보를 구성합니다.
4. 값을 확인하고 "마침"을 클릭합니다.

사용자가 OAuth 권한 부여 서버 자격 증명을 사용하여 페더레이션된 리소스에 싱글 사인온할 수 있도록 OAuth 파트너 관계가 구성됩니다.

페더레이션 시스템이 다음과 같은 형식의 사용자 요청을 받으면 이 요청을 파트너 관계 구성에 따라 처리됩니다.

https://*baseURL_of_the_partnership*/affwebservices/public/oauthtokenconsumer?AuthzServerID=*authorization_server_id*

또는

https://*baseURL_of_the_partnership*/affwebservices/public/oauthtokenconsumer/*disambiguation_id*?AuthzServerID=<*authorization_server_id*>

페더레이션 시스템이 소셜 사인온 기능을 구현하도록 구성됩니다.

OAuth 인증 체계 설정을 OAuth 파트너 관계로 마이그레이션

현재 환경에서 OAuth 공급자를 통해 사용자를 인증하도록 OAuth 인증 체계를 구성한 경우 인증 체계 설정을 페더레이션 파트너 관계로 마이그레이션할 수 있습니다.

다음 단계를 수행하십시오.

1. 다음 단계 중 하나를 수행합니다.
   - OAuth 인증 체계와 OAuth 파트너 관계를 동시에 사용하려는 경우 응용 프로그램을 OAuth 권한 부여 서버에 등록하고 다음 형식의 새 리디렉션 URL 을 기존 OAuth 인증 체계 리디렉션 URL 에 추가합니다.

     `https://server:port/affwebservices/public/oauthtokenconsumer`
   - OAuth 인증 체계 대신 OAuth 파트너 관계를 사용하려는 경우 OAuth 권한 부여 서버에서 기존 리디렉션 URL 을 다음 형식의 해당 파트너 관계 리디렉션 URL 로 업데이트합니다.

     `https://server:port/affwebservices/public/oauthtokenconsumer`

     참고: 인증 체계 리디렉션 URL 을 파트너 관계 리디렉션 URL 로 업데이트한 후에는 OAuth 인증 체계가 작동하지 않습니다.
2. OAuth 클라이언트와 OAuth 권한 부여 서버 간의 파트너 관계를 생성합니다.
3. OAuth 파트너 관계를 시작할 때 다음 URL 을 사용해야 함을 응용 프로그램 사용자에게 알립니다.

   `https://server:port/affwebservices/public/oauthtokenconsumer?AuthzServerID=AuthorizationServerID`

자격 증명 선택기 페이지 구성

사용자가 Facebook 또는 Twitter 등의 아이덴티티 공급자를 선택하여 인증할 수 있도록 파트너 관계를 구성할 수 있습니다. CA SiteMinder for Secure Proxy Server 에 설치된 자격 증명 처리 서비스를 사용하면 사용자 인증을 위해 선택할 수 있는 여러 아이덴티티 공급자가 수록된 자격 증명 선택기 페이지를 표시하도록 파트너 관계를 구성할 수 있습니다.

자격 증명 선택기 페이지를 구성하려면 다음 파트너 관계를 만드십시오.

1. 페더레이션 시스템과 아이덴티티 공급자 사이에서 싱글 사인온을 구성하기 위한 파트너 관계. 아이덴티티 공급자는 어설션 당사자로서 기능하고 페더레이션 시스템은 신뢰 당사자로서 기능합니다.
2. 페더레이션 시스템과 페더레이션된 리소스가 있는 엔터프라이즈 사이의 파트너 관계. 페더레이션 시스템은 어설션 당사자로서 기능하고 엔터프라이즈는 신뢰 당사자로서 기능합니다.

다음 프로세스는 페더레이션 시스템이 사용자 요청을 처리하는 방법을 설명합니다.

1. 엔터프라이즈(신뢰 당사자)는 사용자 요청을 페더레이션 시스템(어설션 당사자)으로 리디렉션합니다.
2. 페더레이션 시스템(어설션 당사자)은 파트너 관계가 자격 증명 선택기 페이지를 표시하도록 구성되었는지 확인합니다. 구성된 경우 사용자 인증을 위해 선택할 수 있는 여러 아이덴티티 공급자가 있는 자격 증명 선택기 페이지가 표시됩니다.
3. 사용자가 페더레이션 시스템에 등록된 경우 다음 단계가 수행됩니다. 사용자가 등록되지 않은 경우 다음 단계로 건너뜁니다.
   a. 사용자가 아이덴티티 공급자를 선택하고 이 아이덴티티 공급자에 사인온합니다.
   b. 아이덴티티 공급자가 액세스 토큰을 생성하고 사용자를 페더레이션 시스템(신뢰 당사자)으로 리디렉션합니다.
   c. 페더레이션 시스템(신뢰 당사자)이 액세스 토큰을 확인하고 사용자 저장소의 사용자를 식별하려고 시도합니다.
   d. 페더레이션 시스템(신뢰 당사자)이 세션을 생성하고 사용자를 페더레이션 시스템(어설션 당사자)으로 리디렉션합니다.
   e. 페더레이션 시스템(어설션 당사자)이 어설션을 생성하고 사용자를 엔터프라이즈(신뢰 당사자)로 리디렉션합니다.
   f. 엔터프라이즈(신뢰 당사자)가 어설션을 확인하고 페더레이션된 리소스에 대한 사용자 액세스를 허가합니다.
4. 사용자가 페더레이션 시스템에 등록되지 않은 경우 다음 단계가 수행됩니다.
   a. 사용자가 "등록" 링크를 클릭합니다.
   b. 페더레이션 시스템이 프로비저닝 서버와 파트너 관계가 구성된 아이덴티티 공급자의 목록을 표시합니다.

c. 사용자가 아이덴티티 공급자를 선택하고 이 아이덴티티 공급자에 사인온합니다.

d. 아이덴티티 공급자가 액세스 토큰을 생성하고 사용자를 페더레이션 시스템(신뢰 당사자)으로 리디렉션합니다.

e. 페더레이션 시스템(신뢰 당사자)이 액세스 토큰을 확인하고 사용자 저장소의 사용자를 식별하려고 시도합니다.

f. 페더레이션 시스템(신뢰 당사자)은 사용자를 파트너 관계에서 구성된 프로비저닝 서버로 사용자를 리디렉션합니다.

g. 프로비저닝 서버가 사용자를 만들어 페더레이션 시스템(신뢰 당사자)으로 리디렉션합니다.

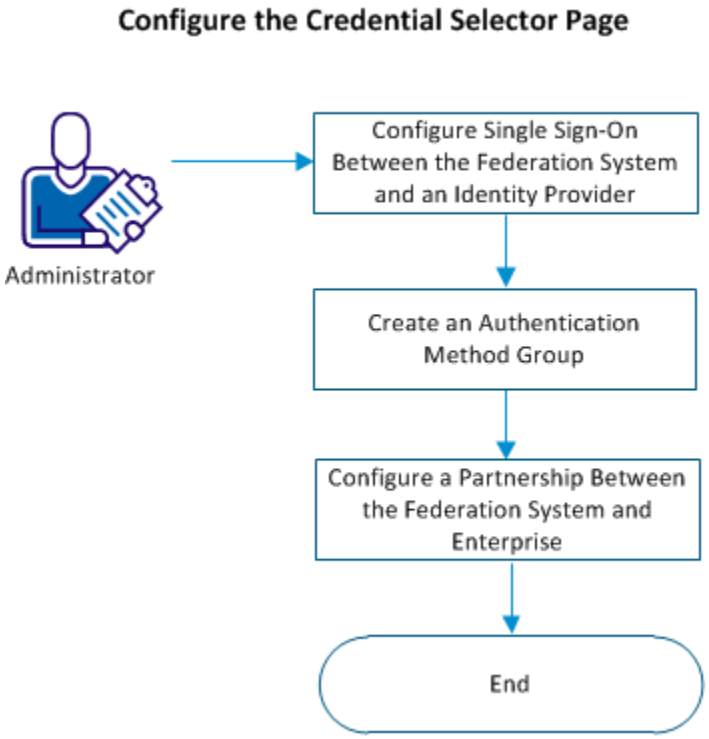
h. 페더레이션 시스템(신뢰 당사자)이 세션을 생성하고 사용자를 페더레이션 시스템(어설션 당사자)으로 리디렉션합니다.

i. 페더레이션 시스템(어설션 당사자)이 어설션을 생성하고 사용자를 엔터프라이즈(신뢰 당사자)로 리디렉션합니다.

j. 엔터프라이즈(신뢰 당사자)가 어설션을 확인하고 페더레이션된 리소스에 대한 사용자 액세스를 허가합니다.

사용자 요청이 처리됩니다.

다음 순서도는 자격 증명 선택기 페이지를 구성하는 방법을 설명합니다.



다음 단계를 수행하십시오.

1. 페더레이션 시스템과 아이덴티티 공급자 사이에서 싱글 사인온을 구성합니다. (페이지 197)
2. 인증 방법 그룹을 만듭니다 (페이지 197).
3. 페더레이션 시스템과 엔터프라이즈 사이에 파트너 관계를 구성합니다 (페이지 198).

페더레이션 시스템과 아이덴티티 공급자 사이에서 싱글 사인온 구성

자격 증명 선택기 페이지에 표시할 각 아이덴티티 공급자에 대해 아이덴티티 공급자와 페더레이션 시스템 사이에 싱글 사인온을 구성하기 위해 파트너 관계를 만드십시오. 아이덴티티 공급자는 어설션 당사자로서 기능하고 페더레이션 시스템은 신뢰 당사자로서 기능합니다.

인증에 사용할 아이덴티티 공급자는 다음 인증 프로토콜에 기반해야 합니다.

- SAML 1.1
- SAML 2.0
- WS-페더레이션
- OAuth

페더레이션 시스템이 아이덴티티 공급자로서 기능하도록 하려면 어설션 당사자와 신뢰 당사자 모두로 기능하는 시스템과 파트너 관계를 만드십시오.

다음 단계를 수행하십시오.

1. "페더레이션", "파트너 관계"로 이동합니다.
2. 자격 증명 선택기 페이지에 표시할 각 아이덴티티 공급자에 대해 파트너 관계를 만듭니다.

인증 방법 그룹 만들기

인증 방법 그룹은 자격 증명 선택기 페이지에 표시되어야 하는 아이덴티티 공급자의 목록을 정의합니다. 자격 증명 선택기 페이지에 표시할 SAML 또는 Facebook 같은 각 아이덴티티 공급자는 인증 방법 그룹의 일부여야 합니다. 인증 방법 그룹을 만들 때는 모든 파트너 관계 목록에서 어설션 당사자로 기능하는 아이덴티티 공급자를 선택할 수 있습니다.

다음 단계를 수행하십시오.

1. "인프라", "인증", "인증 방법 그룹"으로 이동합니다.
2. "인증 방법 그룹 만들기"를 클릭합니다.
3. 인증을 위해 선택할 수 있도록 표시할 아이덴티티 공급자의 파트너 관계를 추가하고 필수 값을 입력합니다.
4. 변경 내용을 저장합니다.

페더레이션 시스템과 엔터프라이즈 사이에 파트너 관계 구성

사용자가 페더레이션된 리소스에 액세스를 시도할 때 자격 증명 선택기 페이지를 표시하려면 페더레이션 시스템과 엔터프라이즈 사이에 파트너 관계를 구성하십시오. 페더레이션 시스템은 어설션 당사자로서 기능하고 엔터프라이즈는 신뢰 당사자로서 기능합니다. 파트너 관계를 만들거나 기존 파트너 관계를 수정할 수 있습니다.

파트너 관계는 다음 인증 프로토콜 중 *하나*에 기반해야 합니다.

- SAML 1.1
- SAML 2.0
- WS-페더레이션

다음 단계를 수행하십시오.

1. "페더레이션", "파트너 관계"로 이동합니다.
2. 각 단계에 값을 입력합니다.
3. "싱글 사인온" 또는 "SSO 및 SLO" 또는 "싱글 사인온 및 사인아웃" 단계에서 다음 단계를 수행합니다.
   a. "자격 증명 선택기"로 "인증 모드"를 선택합니다.
   b. "인증 기준 URL"을 정의합니다.
   c. "인증 방법 그룹"을 선택합니다.
4. "대상 응용 프로그램 구성" 단계에서 다음 필드를 선택합니다.
   - SAML 1.1: 대상
   - SAML 2.0 및 WS-페더레이션: 릴레이 상태가 대상 무시
5. 변경 내용을 저장합니다.

사용자가 페더레이션된 리소스에 액세스를 시도하면 자격 증명 선택기를 표시하도록 파트너 관계가 구성되었습니다.

페더레이션 시스템이 소셜 사인온 기능을 구현하도록 구성됩니다.

자격 증명 선택기 페이지에서 머리글 및 바닥글 사용자 지정

엔터프라이즈 요건에 맞게 자격 증명 선택기 페이지에 표시되는 머리글과 바닥글을 사용자 지정할 수 있습니다.

다음 단계를 수행하십시오.

1. 페더레이션 시스템에서 다음 위치로 이동합니다.

   <*install_path*>\CA\secure-proxy\Tomcat\webapps\chs\jsps

2. header.jsp 파일의 복사본을 만든 다음 새 파일의 이름을 header-custom.jsp 로 지정합니다.
3. footer.jsp 파일의 복사본을 만든 다음 새 파일의 이름을 footer-custom.jsp 로 지정합니다.

   참고: header-custom.jsp 및 footer-custom.jsp 파일이 있으면 머리글 및 바닥글 표시에 이 파일을 사용하도록 페더레이션 시스템이 구성됩니다.
4. 이 파일을 수정하여 자격 증명 선택기 페이지에 표시되어야 하는 머리글과 바닥글을 사용자 지정합니다.
5. 변경 내용을 저장합니다.
6. CA SiteMinder for Secure Proxy Server 를 다시 시작합니다.

   파트너 관계가 활성화되었을 때 사용자 지정된 머리글 및 바닥글이 자격 증명 선택기 페이지에 표시됩니다.

제 13 장: 어설션 처리 사용자 지정(신뢰 당사자)

메시지 소비자 플러그인은 메시지 소비자 확장 API 를 구현하는 Java 프로그램입니다. 이 플러그인을 통해 어설션 거부, 상태 코드 반환 등의 어설션 처리를 위한 사용자 고유의 비즈니스 논리를 구현할 수 있습니다. 이 추가 처리는 어설션의 표준 처리와 함께 작동합니다.

인증이 진행되는 동안 시스템은 먼저 사용자를 로컬 사용자 저장소에 매핑하여 어설션을 처리하려고 합니다. CA SiteMinder?Federation 이 사용자를 찾을 수 없는 경우 메시지 소비자 플러그인의 postDisambiguateUser 메서드를 호출합니다.

플러그인이 사용자를 찾은 경우 인증의 두 번째 단계로 진행합니다. 플러그인이 사용자를 로컬 사용자 저장소에 매핑할 수 없는 경우에는 UserNotFound 오류가 반환됩니다. 플러그인이 선택적으로 리디렉션 URL 기능을 사용할 수 있습니다. 소비자 플러그인이 없는 경우 리디렉션 URL 은 SAML 인증 체계가 생성하는 오류를 기반으로 합니다.

두 번째 인증 단계에서 시스템은 플러그인이 구성된 경우 메시지 소비자 플러그인의 postAuthenticateUser 메서드를 호출합니다. 메서드가 성공하는 경우 CA SiteMinder?Federation 은 사용자를 요청된 리소스로 리디렉션합니다. 메서드가 실패하는 경우 사용자를 실패 페이지에 보내도록 플러그인을 구성할 수 있습니다. 실패 페이지는 인증 체계 구성으로 지정할 수 있는 리디렉션 URL 중 하나일 수 있습니다.

참조 정보(메서드 서명, 매개 변수, 반환 값, 데이터 형식)와 UserContext 클래스에 대한 생성자는 *Java SDK 프로그래밍 참조서*에 나와 있습니다. MessageConsumerPlugin 인터페이스를 참조하십시오.

플러그인을 구성하려면

1. CA SiteMinder® Federation SDK 를 설치합니다.
2. SDK 의 일부인 MessageconsumerPlugin.java 인터페이스를 구현합니다.
3. 메시지 소비자 플러그인 구현 클래스를 배포합니다.
4. 관리 UI 에서 메시지 소비자 플러그인이 사용되도록 설정합니다.

MessageConsumerPlugin 인터페이스 구현

MessageConsumerPlugin.java 인터페이스를 구현하여 사용자 지정 메시지 소비자 플러그인을 생성하십시오. 다음 절차에는 구현 클래스에 대한 최소 요구 사항이 나열되어 있습니다.

다음 단계를 수행하십시오.

1. 매개 변수가 포함되지 않은 공개 기본 생성자 메서드를 제공합니다.
2. 상태 비저장 구현이 되도록 코드를 제공합니다. 여러 스레드가 단일 플러그인 클래스를 사용할 수 있어야 합니다.
3. 인터페이스에서 요구 사항을 충족할 메서드를 구현합니다.

   MessageConsumerPlugin 에는 다음 네 가지 메서드가 포함됩니다.

   init()

   플러그인에 필요한 시작 절차를 수행합니다. SiteMinder 는 플러그인이 로드될 때 각 플러그인 인스턴스에 대해 한 번씩 이 메서드를 호출합니다.

   release()

   플러그인에 필요한 런다운 절차를 모두 수행합니다. SiteMinder 는 SiteMinder 가 종료될 때 각 플러그인 인스턴스에 대해 한 번씩 이 메서드를 호출합니다.

   postDisambiguateUser()

   인증 체계가 사용자 명확성 처리를 수행할 수 없을 때 해당 처리를 제공합니다. 또는 이 메서드가 새 페더레이션 사용자에 대한 데이터를 사용자 저장소에 추가할 수 있습니다. 이 메서드는 암호 해독된 어설션을 수신합니다. 암호 해독된 어설션은 플러그인에 전달된 속성 맵의 "_DecryptedAssertion" 키 아래에 추가됩니다.

   postAuthenticateUser()

   정책 서버 처리 성공 여부와 관계없이 어설션 처리 결과를 확인하기 위한 추가 코드를 제공합니다.

제품은 다음과 같은 메시지 소비자 플러그인 클래스 샘플을 제공합니다.

- MessageConsumerPluginSample.java
- MessageConsumerSAML20.java

샘플의 기본 위치는 다음과 같습니다.

Windows

C:\Program Files\FederationManager\sdk\java\sample

패키지 이름은 com\ca\federation\sdk\plugin\sample 입니다.

UNIX

/FederationManager/sdk/java/sample

패키지 이름은 com/ca/federation/sdk/plugin/sample 입니다.

메시지 소비자 플러그인 배포

MessageConsumerPlugin 인터페이스에 대한 구현 클래스를 코드로 지정한 다음에는 해당 구현 클래스를 컴파일하고 CA SiteMinder?Federation 이 실행 파일을 찾을 수 있는지 확인하십시오.

다음 단계를 수행하십시오.

1. MessageConsumerPlugin Java 파일을 컴파일합니다. 이 파일을 컴파일하려면 제품과 함께 설치되는 다음 종속 라이브러리가 필요합니다.

   federation_install_dir\siteminder\bin\jars\SmJavaApi.jar

   federation_install_dir 은 CA SiteMinder® Federation 이 설치된 디렉터리입니다.

2. 폴더나 jar 파일에서 플러그인 클래스를 사용할 수 있는 경우 JVMOptions.txt 파일에서 -Djava.class.path 값을 수정합니다. 이 단계를 수행하면 수정된 클래스 경로를 사용하여 플러그인 클래스를 로드할 수 있습니다.

   federation_mgr_installation_home\siteminder\config 디렉터리에서 JVMOptions.txt 파일을 찾습니다.

   참고: 기존 xerces.jar, xalan.jar 또는 SmJavaApi.jar 의 클래스 경로를 수정하지 마십시오.

3. 시스템을 다시 시작하여 최신 버전의 MessageConsumerPlugin 을 선택합니다. 이 단계는 플러그인 Java 파일이 다시 컴파일될 때마다 필요합니다.
4. 플러그인이 사용되도록 설정합니다.

UI 에서 메시지 소비자 플러그인이 사용되도록 설정

메시지 소비자 플러그인을 작성하고 컴파일한 후 관리 UI 에서 설정을 구성하여 플러그인이 사용되도록 설정하십시오. UI 설정은 CA SiteMinder?Federation 에게 플러그인을 찾을 수 있는 위치를 알려 줍니다.

플러그인을 배포 (페이지 203)할 때까지 플러그인 설정을 구성하지 마십시오.

메시지 소비자 플러그인이 사용되도록 설정하려면

1. 관리 UI 에 로그온합니다.

   수정할 소비자-생산자 또는 SP-IdP 파트너 관계를 선택합니다.
2. 파트너 관계 마법사의 "사용자 ID" 단계로 이동합니다.
3. "메시지 소비자 플러그인" 섹션의 다음 필드에 데이터를 입력합니다.

   플러그인 클래스

   플러그인에 대한 Java 클래스 이름을 지정합니다. 예를 들어 SDK 에 포함된 샘플 클래스는 다음과 같습니다.

   com.ca.messageconsumerplugin.MessageConsumerPluginSample

   플러그인 매개 변수

   "전체 Java 클래스 이름" 필드에서 지정한 플러그인에 전달되는 매개 변수 문자열을 지정합니다.

4. 운영 환경에 맞게 페더레이션 서비스를 다시 시작합니다.

   - **Windows**

     다음과 같이 중지 및 시작 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

     a. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"

     b. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"

   - **UNIX**

     a. 명령 창을 엽니다.

     b. 다음 스크립트를 실행합니다.

     *federation_install_dir/*fedmanager.sh stop

     federation_install_dir/fedmanager.sh start

     참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

제 14 장: 위임된 인증

위임된 인증 개요

싱글 사인온에 대한 구성 결정 사항 중 하나는 사용자 인증 방법의 결정입니다.

SiteMinder 에서는 다음 두 가지 인증 방법을 제공합니다.

- 로컬 인증

  SiteMinder 가 사용자를 로컬 사이트에서 인증합니다. 사용자가 인증으로 리디렉션되고 세션을 설정하는 관리 UI 에서 인증 URL 을 구성합니다.

- 위임된 인증

  SiteMinder 가 SiteMinder 에서 보호하지 않는 타사 WAM(웹 액세스 관리) 응용 프로그램을 사용합니다. 타사 응용 프로그램은 보호된 페더레이션된 리소스를 요청하는 모든 사용자를 인증한 다음 페더레이션된 사용자 아이덴티티를 SiteMinder 로 전달합니다. SiteMinder 는 사용자 아이덴티티 정보를 수신한 후 자체 사용자 디렉터리에서 사용자를 찾고 신뢰 당사자와 페더레이션 프로세스를 시작합니다.

위임된 인증 요청은 어설션 당사자 측에서 수행되며 타사 WAM 시스템 또는 SiteMinder 에서 시작될 수 있습니다. 인증 요청은 신뢰 당사자 측에서 시작될 수 있지만 이 시나리오는 위임된 인증으로 간주되지 않습니다.

인증은 다음과 같이 시작될 수 있습니다.

어설션 당사자 측에서 SiteMinder 의해 시작되는 인증

SiteMinder 는 어설션 당사자 측에서 인증 요청을 시작할 수 있습니다. 요청이 SiteMinder 로 전송되면 이 요청은 위임된 인증 요청으로 인식됩니다. 그러면 SiteMinder 는 사용자를 타사 WAM 시스템으로 리디렉션합니다.

어설션 당사자 측에서 WAM 시스템에 대한 직접 로그인으로 시작된 인증

사용자가 어설션 당사자 측에서 WAM 시스템에 로그인하면 인증 요청이 시작됩니다. WAM 시스템이 사용자를 성공적으로 인증하면 아이덴티티 정보가 SiteMinder 로 전달됩니다.

신뢰 당사자 측에서 시작된 인증

신뢰 당사자는 인증 요청을 시작할 수 있지만 이 시나리오는 위임된 인증으로 간주되지 않습니다. 위임된 인증은 어설션 당사자 측에서만 수행됩니다.

페더레이션된 리소스에 대한 요청은 신뢰 당사자로 직접 전송되며 신뢰 당사자는 AuthnRequest 를 어설션 당사자의 SiteMinder 로 전송합니다. SiteMinder 는 이를 위임된 인증 요청으로 인식하고 사용자를 어설션 당사자의 타사 WAM 시스템으로 리디렉션합니다. 사용자는 WAM 시스템에 로그인하고 이 시스템이 인증 요청을 시작합니다. WAM 시스템이 사용자를 성공적으로 인증하면 아이덴티티 정보가 SiteMinder 로 전달됩니다.

타사 WAM 시스템은 인증 요청을 받은 후 사용자 아이덴티티를 SiteMinder 로 전달합니다. WAM 시스템이 사용자 아이덴티티를 전달하는 데 사용하는 방법은 위임된 인증 방법이 쿠키 기반인지 쿼리 문자열 기반인지에 따라 달라집니다.

타사 WAM 이 사용자 아이덴티티를 전달하는 방법

타사 WAM 시스템은 다음 두 방법 중 하나를 사용하여 페더레이션된 사용자 아이덴티티를 SiteMinder 로 전달합니다.

- 개방 형식 쿠키를 사용합니다.

  데이터의 보안을 위하여 개방 형식 쿠키를 암호화할 수 있습니다.

- 브라우저를 SiteMinder 로 보내는 리디렉션 URL 에 추가되는 쿼리 문자열을 사용합니다.

  쿼리 문자열은 ClearText 로 전송됩니다.

  중요! 프로덕션 환경에서는 쿼리 문자열 방법을 사용하지 마십시오. 쿼리 문자열 리디렉션 방법은 테스트 환경에서 개념 증명용으로만 사용해야 합니다.

타사 WAM 시스템이 선택하는 방법은 사용자 아이덴티티를 SiteMinder 로 전달하기 위해 시스템에서 설정하려는 구성에 따라 달라집니다.

사용자 아이덴티티를 전달하는 방법은 다음 단원에서 자세히 설명합니다.

사용자 아이덴티티를 전달하기 위한 쿠키 방법

SiteMinder 는 개방 형식 쿠키를 사용하여 사용자 아이덴티티를 전달할 수 있습니다. 쿠키에는 사용자 로그인 ID 가 값의 하나로 포함됩니다.

인증은 WAM 시스템 또는 SiteMinder 에서 시작될 수 있습니다. 인증이 SiteMinder 에서 시작되는 경우에는 사용자가 WAM 시스템으로 리디렉션됩니다. 인증 프로세스는 WAM 시스템에서 시작될 때와 동일합니다.

위임된 인증 프로세스는 다음과 같습니다.

1. 인증 요청이 타사 WAM 시스템에 수신됩니다.
2. 사용자가 인증됩니다.
3. 타사 WAM 시스템은 다음 두 방법 중 하나로 쿠키를 가져옵니다.
   - WAM 시스템이 CA SiteMinder® Federation SDK 를 사용하여 개방 형식 쿠키를 생성합니다. SDK 가 쿠키를 생성하여 WAM 시스템에 대한 요청으로 다시 전송합니다.

     참고: FIPS 암호화 개방 형식 쿠키를 생성하려면 CA SiteMinder® Federation SDK 를 사용하십시오.

     타사 WAM 응용 프로그램은 쿠키를 생성하기 위해 사용하는 SDK 와 동일한 언어를 사용합니다. CA SiteMinder® Federation Java SDK 를 사용하는 경우 타사 WAM 응용 프로그램은 Java 로 작성되어야 합니다. .NET SDK 를 사용하는 경우 타사 WAM 응용 프로그램이 .NET 을 지원해야 합니다.
   - WAM 시스템은 수동으로 생성된 개방 형식 쿠키를 사용합니다.

     CA SiteMinder® Federation SDK 를 사용하지 않고 개방 형식 쿠키를 생성할 수 있습니다. 쿠키를 수동으로 생성하려면 UTF-8 인코딩을 지원하는 프로그래밍 언어를 사용하십시오. 암호 기반 암호화에 대해 SiteMinder 가 지원하는 다음 PBE 암호화 알고리즘을 사용할 수 있습니다.
     - PBE/SHA1/AES/CBC/PKCS12PBE-1000-128
     - PBE/SHA1/AES/CBC/PKCS12PBE-1000-192
     - PBE/SHA1/AES/CBC/PKCS12PBE-1000-256
     - PBE/SHA256/AES/CBC/PKCS12PBE-1000-128

- PBE/SHA256/AES/CBC/PKCS12PBE-1000-192
- PBE/SHA256/AES/CBC/PKCS12PBE-1000-256
- PBE/SHA1/3DES_EDE/CBC/PKCS12PBE-1000-3
- PBE/SHA256/3DES_EDE/CBC/PKCS12PBE-1000-3

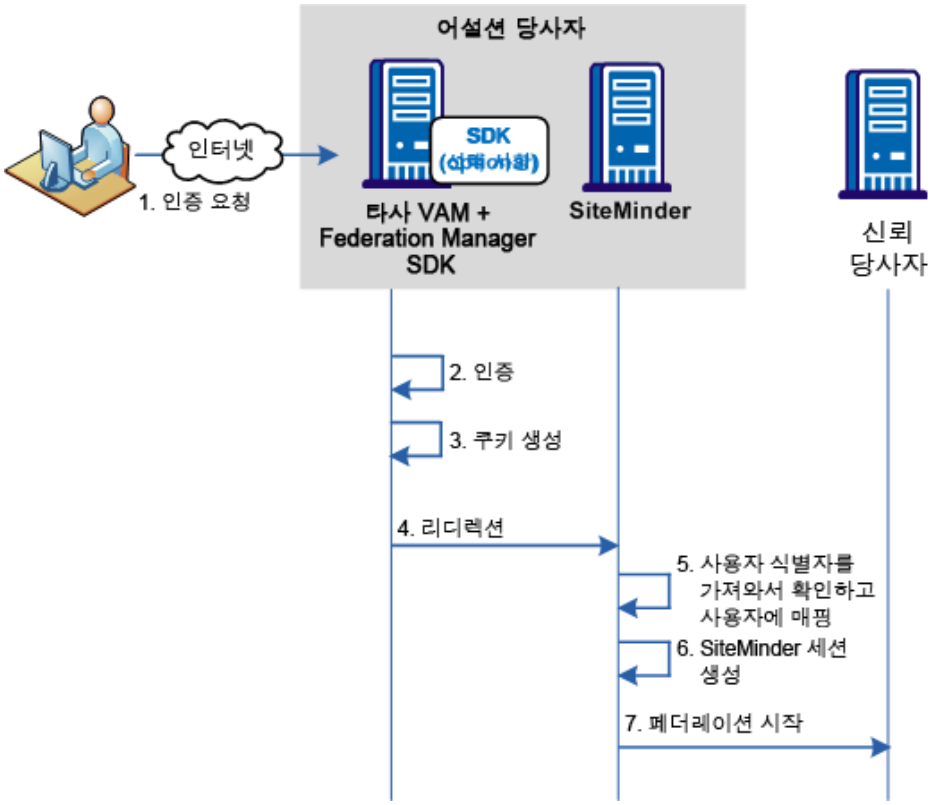
브라우저에서 개방 형식 쿠키가 설정되었는지 확인하십시오.

완전한 쿠키를 작성하려면 개방 형식 쿠키의 콘텐츠에 대한 자세한 정보를 참조하십시오.

참고: WAM 시스템과 SiteMinder 는 동일한 쿠키 도메인에 있어야 합니다.

4. WAM 시스템이 브라우저를 SiteMinder 로 리디렉션합니다.
5. SiteMinder 는 쿠키에서 로그인 ID 를 추출한 다음 자체 사용자 디렉터리에서 사용자를 찾습니다.
6. SiteMinder 가 SiteMinder 세션을 생성합니다.
7. 세션이 생성되면 신뢰 당사자와의 페더레이션 통신이 진행됩니다.

다음 그림은 인증이 타사 WAM 에서 시작되었을 때의 쿠키 방법을 보여 줍니다. SiteMinder 는 WAM 응용 프로그램을 보호하고 있지 않습니다.



중요! SDK 로 만든 개방 형식 쿠키를 사용하려면 타사는 CA SiteMinder® Federation SDK 를 설치해야 합니다. SDK 는 SiteMinder 와 별도로 설치되는 구성 요소입니다. 설치 키트에는 위임된 인증에 SDK 를 사용하는 방법을 설명하는 문서가 포함되어 있습니다.

사용자 아이덴티티를 전달하기 위한 쿼리 문자열 방법

타사 WAM 시스템은 리디렉션 URL 에 쿼리 문자열을 추가하여 사용자 아이덴티티를 SiteMinder 에 전달할 수 있습니다. 이 방법이 작동하려면 타사 WAM 시스템이 페더레이션된 사용자가 인증된 후에 이러한 사용자를 SiteMinder 로 리디렉션하는 URL 을 구성해야 합니다.

중요! 프로덕션 환경에서는 쿼리 문자열 방법을 사용하지 마십시오. 쿼리 문자열 리디렉션 방법은 테스트 환경에서 개념 증명용으로만 사용해야 합니다.

인증이 WAM 시스템에서 시작되는 경우 쿼리 문자열을 사용하는 위임된 인증의 프로세스는 다음과 같습니다.

참고: 인증은 SiteMinder 또는 신뢰 당사자 측에서 시작될 수도 있습니다.

1. 타사 WAM 시스템이 인증 요청을 수신합니다.
2. 사용자가 인증됩니다.
3. 타사 WAM 시스템은 리디렉션 URL 을 구성하고 로그인 ID 와 해시된 로그인 ID 값을 LoginID=*LoginID*&LoginIDHash=*hashed_LoginID* 의 형식으로 쿼리 문자열에 추가합니다.

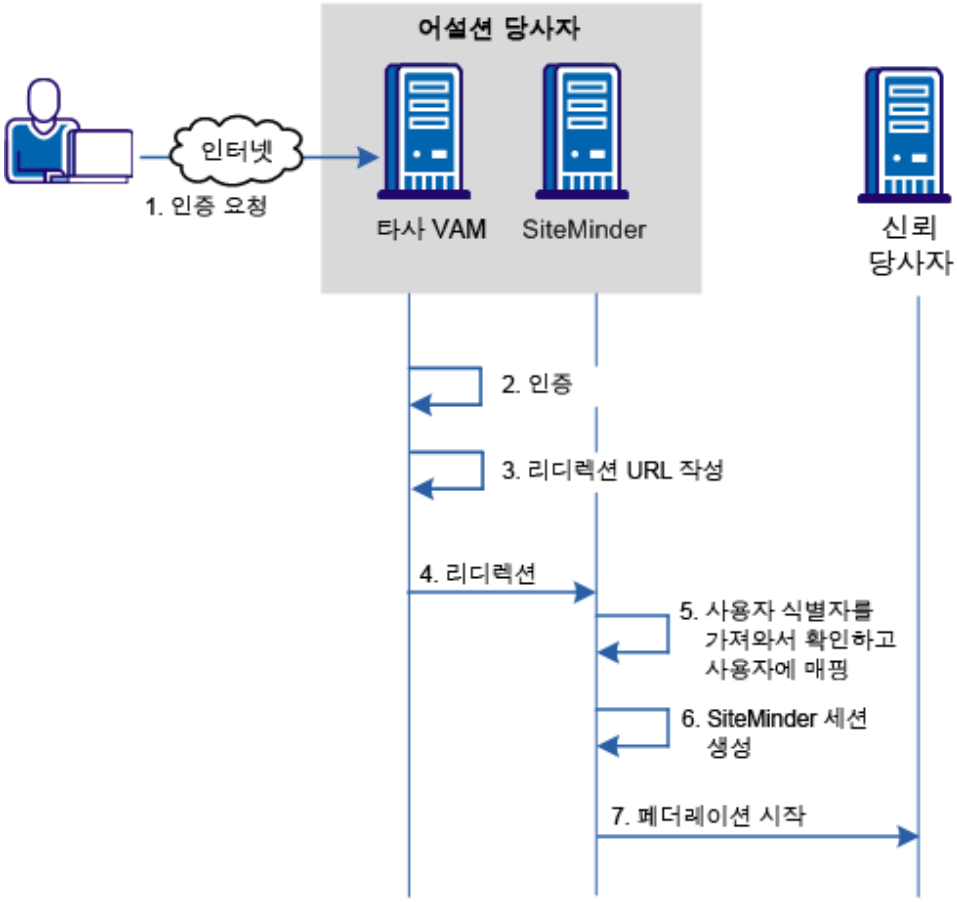
   중요! LoginID 및 LoginIDHash 매개 변수는 대/소문자를 구분합니다. 이 매개 변수를 예제에 나오는 그대로 리디렉션 URL 에 포함하십시오.

   SiteMinder 는 해싱 메커니즘을 통해 사용자 ID 가 변경되지 않고 수신되었음을 확인할 수 있습니다.

   리디렉션 URL 의 예

   http://idp1.example.com:9090/affwebservices/public/saml2sso?SPID=FmSP&ProtocolBinding=urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST&LoginID=jdoe&LoginIDHash=454d3bd5cb839168eeffcf060ae0b9c28ed6eec0
4. WAM 시스템이 브라우저를 SiteMinder 로 리디렉션합니다.
5. SiteMinder 는 URL 에서 로그인 ID 와 해시된 로그인 ID 를 추출하고 해시된 값을 사용하여 식별자를 확인한 다음 자체 사용자 디렉터리에서 사용자를 찾습니다.
6. SiteMinder 가 사용자 세션을 만듭니다.
7. 세션이 생성되면 신뢰 당사자와의 페더레이션 통신이 진행됩니다.

다음 그림은 인증이 어설션 당사자 측에서 시작될 때의 쿼리 문자열 방법을 보여 줍니다.



위임된 인증 구성

위임된 인증은 인증된 사용자 아이덴티티를 기반으로 어설션이 생성되는 어설션 당사자 측에서 구성됩니다.

위임된 인증을 구성하려면

1. 타사 WAM 이 사용자 아이덴티티를 전달하기 위해 사용하는 방법(쿠키 또는 쿼리 문자열)을 확인합니다.

   참고: 쿼리 문자열은 FIPS 호환 파트너 관계를 생성하지 않습니다.

2. 파트너 관계 마법사의 적절한 단계로 이동하여 위임된 인증을 설정합니다.

중요! SDK 로 생성한 개방 형식 쿠키를 사용하려면 타사에서 CA SiteMinder® Federation SDK 를 설치해야 합니다. SDK 는 별도로 설치되는 구성 요소입니다. 설치 키트에는 위임된 인증에 SDK 를 사용하는 방법을 설명하는 문서가 포함되어 있습니다.

쿠키 위임된 인증 샘플 설정

다음 샘플 구성은 SAML 2.0 IdP > SP 파트너 관계 관점에서의 샘플입니다. 위임된 인증 설정은 파트너 관계 마법사의 SSO 및 SLO 단계에 있습니다.

이 샘플 구성은 SAML 2.0 구성을 반영합니다. 아이덴티티 공급자는 http://idp1.xyz.com 이며 타사 WAM 시스템은 http://wamservice.xyz.com 입니다.

쿠키 위임된 인증을 구성하려면

1. 파트너 관계를 생성하거나 기존 파트너 관계를 편집합니다.

   참고: 파트너 관계를 편집하려면 먼저 비활성화하십시오.

2. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.

3. "인증" 섹션에서 다음과 같이 필드를 설정합니다.

   인증 모드

   위임됨

   위임된 인증 유형

   개방 형식 쿠키

   웹 액세스 관리 응용 프로그램에 사용하기 위한 것입니다. CA SiteMinder® Federation SDK 를 사용하여 Java 또는 .NET 응용 프로그램을 생성할 수 있습니다. 또는 개방 형식 쿠키를 수동으로 만든 경우 다른 언어로 작성된 응용 프로그램을 사용할 수 있습니다.

   FIPS 140-2 암호화가 필요한 경우 CA SiteMinder® Federation Java 또는 .NET SDK 를 사용하여 개방 형식 쿠키를 만드십시오.

   위임된 인증 URL

   http://wamservice.xyz.com

   사용자를 인증하고 CA SiteMinder® Federation SDK 를 사용하여 쿠키를 만드는 타사 WAM 시스템의 URL 입니다.

   인증 클래스

   타사에서 사용되는 인증 방법을 입력합니다. 예를 들면 다음과 같습니다.

   urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos

4. 모든 개방 형식 쿠키 설정을 타사 WAM 시스템으로 전달합니다.

   SiteMinder 는 쿠키를 생성할 때 이 값을 사용합니다.

5. 파트너 구성을 계속합니다.

쿼리 문자열 위임된 인증 샘플 설정

다음 샘플 구성은 SAML 2.0 IdP > SP 파트너 관계 관점에서의 샘플입니다. 위임된 인증 설정은 파트너 관계 마법사의 SSO 및 SLO 단계에 있습니다.

참고: 쿼리 문자열 방법은 FIPS 호환 파트너 관계를 생성하지 않습니다.

이 샘플 구성은 SAML 2.0 구성을 반영합니다. 아이덴티티 공급자는 http://idp1.xyz.com 이며 타사 WAM 시스템은 http://wamservice.xyz.com 입니다.

중요! 프로덕션 환경에서는 쿼리 문자열 방법을 사용하지 마십시오. 쿼리 문자열 리디렉션 방법은 테스트 환경에서 개념 증명용으로만 사용해야 합니다.

쿼리 문자열 위임된 인증을 구성하려면

1. 파트너 관계를 생성하거나 기존 파트너 관계를 편집합니다.

   참고: 파트너 관계를 편집하려면 먼저 비활성화하십시오.

2. 파트너 관계 마법사의 적절한 단계로 이동합니다.
3. "인증" 섹션에서 다음과 같이 필드를 설정합니다.

   인증 모드

   위임됨

   위임된 인증 유형

   쿼리 문자열

   위임된 인증 URL

   http://wamservice.xyz.com

   사용자를 인증하고 쿼리 매개 변수가 포함된 SiteMinder 로의 리디렉션 URL 을 구성하는 타사 WAM 시스템의 URL 입니다.

   해시 암호

   FederatedAuth1

   타사 WAM 시스템은 이 암호를 사용하여 로그인 ID 를 해시합니다.

   해시 암호 확인

   FederatedAuth1

   인증 클래스

   타사에서 사용되는 인증 방법을 입력합니다. 예를 들면 다음과 같습니다.

   urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos

4. 파트너 구성을 계속합니다.

쿠키 위임된 인증에 대한 타사 WAM 구성

위임된 인증에 성공하려면 타사 WAM 이 다음과 같이 페더레이션된 응용 프로그램을 조정해야 합니다.

- 인증된 사용자 로그인 ID 를 쿠키를 통해 통신하려면 타사 WAM 시스템이 쿠키를 생성해야 합니다.
  - Java 응용 프로그램의 경우 WAM 은 CA SiteMinder® Federation Java SDK 를 사용하여 레거시 쿠키 또는 개방 형식 쿠키를 생성할 수 있습니다.
  - .NET 응용 프로그램의 경우 WAM 은 CA SiteMinder® Federation .NET SDK 를 사용하여 개방 형식 쿠키를 생성할 수 있습니다.
  - Java 및 .NET 이외의 언어인 경우 WAM 은 개방 형식 쿠키를 수동으로 생성할 수 있습니다.

  필요한 클래스 및 방법의 구현에 대한 자세한 내용은 *CA SiteMinder® Federation Java SDK 안내서* 또는 *CA SiteMinder® Federation .NET SDK 안내서* 를 참조하십시오.각 안내서는 SDK 와 함께 설치됩니다. 개방 형식 쿠키를 수동으로 생성하는 경우 쿠키의 필수 콘텐츠에 대한 자세한 내용을 검토하십시오.
- 타사는 SiteMinder 어설션 당사자 측에서 구성되는 다음 관리 UI 설정의 값을 알아야 합니다.
  - 암호화 암호
  - 개방 형식 쿠키 이름
  - 개방 형식 쿠키 암호화 변환

  SiteMinder 는 쿠키를 생성할 때 이 값을 사용합니다. 이 설정은 파트너 관계 마법사의 "싱글 사인온"(SAML 1.x) 단계와 "SSO 및 SLO"(SAML 2.0) 단계에 있습니다.
- 타사 WAM 시스템은 사용자를 다시 SiteMinder 로 보내는 리디렉션 URL 을 만들어야 합니다. 이 URL 은 사용자를 다시 SiteMinder 싱글 사인온 서비스로 보내야 합니다. SiteMinder 관리자는 대역 외 통신을 통해 타사에 이 URL 에 대한 정보를 알려야 합니다.

  중요! 타사 WAM 시스템은 SiteMinder 에서 인증 요청을 수신한 후 기존 쿼리 문자열을 캡처하고 다시 전송해야 합니다. 들어오는 요청의 쿼리 문자열 내에 SiteMinder 요청 정보가 있을 수 있으며 요청은 변경 없이 전달되어야 합니다.

  참고: 쿠키를 전달하려면 타사 WAM 시스템은 어설션 당사자의 SiteMinder 와 동일한 쿠키 도메인에 있어야 합니다.

쿼리 문자열 인증에 대한 타사 WAM 구성

타사 WAM 시스템과 어설션 당사자의 SiteMinder 는 쿼리 문자열을 통해 로그인 ID 를 전달합니다. WAM 시스템은 리디렉션 URL 에서 쿼리 문자열에 다음 두 개의 특성을 추가해야 합니다.

LoginID

사용자를 타사 WAM 시스템에 식별하는 값을 지정합니다.

중요! LoginID 매개 변수는 대/소문자를 구분합니다.

LoginIDHash

LoginID 의 해시입니다.

LoginIDHash 값을 생성할 때는 LoginID 가 해시 암호에 추가된 다음 전체 값에 SHA-1 해싱 알고리즘이 적용됩니다. 해시 암호는 어설션 당사자의 SiteMinder 구성에서 지정됩니다.

SiteMinder 는 쿼리 문자열에서 자격 증명을 검색할 때 이 값을 결합하고 해시합니다. 해시가 동일하면 SiteMinder 는 로그인 ID 를 유효한 것으로 간주하고 페더레이션 요청을 계속 진행합니다.

중요! LoginIDHash 매개 변수는 대/소문자를 구분합니다.

타사 WAM 시스템은 페더레이션된 응용 프로그램을 구성하여 사용자를 다시 SiteMinder 싱글 사인온 서비스로 보내는 리디렉션 URL 을 구성해야 합니다. 따라서 SiteMinder 관리자는 대역 외 통신을 통해 싱글 사인온 서비스를 타사로 전달해야 합니다.

중요! 타사 WAM 시스템은 SiteMinder 에서 인증 요청을 수신한 후에 기존 쿼리 문자열을 캡처하고 다시 전송합니다. 들어오는 요청의 쿼리 문자열 내에 SiteMinder 요청 정보가 있는 경우 WAM 시스템은 이를 변경 없이 전달해야 합니다.

쿼리 문자열의 구문은 다음과 같습니다.

?*existing_query_string*&LoginID=*LoginID*&LoginIDHash=*hashed_LoginID*

예

```
https://johndoe3227.b.com/affwebservices/public/saml2sso?SPID=sp1&
LoginID=user1&LoginIDHash=de164152ed6e8e9a7f760e47d135ecf0c98a
3e4e&ProtocolBinding=urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact
```

제 15 장: 싱글 사인온을 시작하기 위한 URL

싱글 사인온을 시작하는 서블릿에 대한 링크

페더레이션된 콘텐츠의 사이트를 설계할 경우 해당 사이트에는 싱글 사인온을 트리거하는 특정 링크가 있는 페이지가 포함됩니다. 이러한 링크는 싱글 사인온 서비스 또는 AuthnRequest 서비스에 대한 서블릿의 URL 입니다.

싱글 사인온을 시작하려면 사용자는 어설션 당사자 또는 신뢰 당사자 측에서 시작할 수 있습니다. 각 사이트에서 적절한 링크를 구성하여 싱글 사인온 작업을 시작합니다.

생산자에서 시작되는 SSO(SAML 1.1)

생산자에서 사용자를 소비자 사이트에 연결하는 링크가 포함된 페이지를 생성하십시오. 각 링크는 사이트 간 전송 URL 을 나타냅니다. 사용자는 사이트 간 전송 URL 을 방문해야 합니다. URL 은 사용자가 소비자 사이트로 리디렉션되기 전에 생산자 측 웹 에이전트에 요청을 보냅니다.

SAML 아티팩트 및 POST 프로필의 경우 사이트 간 전송 URL 의 구문은 다음과 같습니다.

```
http://producer_host:port/affwebservices/public/intersitetransfer?
CONSUMERID=consumer_entity_ID&TARGET=http://consumer_site/target_url
```

이 사이트 간 전송 URL 의 변수 및 쿼리 매개 변수는 다음과 같습니다.

producer_host:port

사용자가 인증되는 서버 및 포트 번호를 지정합니다.

CONSUMERID

(필수) 소비자를 식별합니다. 생산자 측에서 생산자-소비자 파트너 관계에는 이름이 있고 원격 소비자 엔터티에는 ID 가 있습니다. CONSUMERID 는 원격 소비자의 엔터티 ID 입니다. 엔터티 ID 는 대/소문자를 구분합니다. "관리 UI"에 표시되는 그대로 입력하십시오.

CONSUMERID 대신 매개 변수 NAME 을 사용할 수 있지만 둘 다 사용할 수는 없습니다.

NAME 을 사용할 때는 생산자에서 정의된 생산자-소비자 파트너 관계의 이름을 지정하십시오.

consumer_entity_ID

사용자가 생산자 사이트에서 방문하려는 소비자 사이트를 나타냅니다. 엔터티 ID 는 대/소문자를 구분합니다. "관리 UI"에 표시되는 그대로 입력하십시오.

TARGET

(선택 사항) 소비자에서 요청된 대상 리소스를 나타냅니다.

TARGET 매개 변수는 선택 사항입니다. 대상을 정의해야 하지만 사이트 간 전송 URL 이 아니라 소비자 측 파트너 관계에서 정의할 수 있습니다. 대상은 파트너 관계 마법사의 "응용 프로그램 통합" 단계에서 정의됩니다. URL 또는 파트너 관계에서 대상을 정의하십시오.

consumer_site

소비자 측에서 서버를 지정합니다.

target_url

소비자 측의 대상 응용 프로그램을 지정합니다.

참고: SAML 아티팩트 바인딩에 대한 쿼리 매개 변수는 HTTP-인코딩을 사용해야 합니다.

아티팩트 및 POST 프로필에 대한 사이트 간 전송 URL 의 예는 다음과 같습니다.

```
http://www.smartway.com/affwebservices/public/intersitetransfer?
CONSUMERID=ahealthco&TARGET=http://www.ahealthco.com:85/
smartway/index.jsp
```

IdP 에서 시작되는 SSO(SAML 2.0 아티팩트 또는 POST)

사용자가 서비스 공급자로 가기 전에 SiteMinder 아이덴티티 공급자를 방문할 경우에는 아이덴티티 공급자에서 원치 않는 응답이 시작되어야 합니다. 원치 않는 응답을 시작하려면 SiteMinder 가 허용하는 HTTP Get 요청을 생성하는 하드 코딩된 링크를 생성하십시오. 이 HTTP Get 요청에는 서비스 공급자 ID 를 제공하는 쿼리 매개 변수가 포함되어야 합니다. 아이덴티티 공급자는 SAML 어설션 응답을 생성해야 합니다. 사용자는 이 링크를 클릭하여 원치 않는 응답을 시작합니다.

참고: 이 정보는 아티팩트 또는 POST 바인딩에 적용됩니다.

원치 않는 응답에서 아티팩트 또는 POST 프로필이 사용되도록 지정하려는 경우 원치 않는 응답 링크의 구문은 다음과 같습니다.

http://*idp_server:port*/affwebservices/public/saml2sso?SPID=*SP_ID*&ProtocolBinding=*URI_for_binding*&RelayState=*target_URL*

idp_server:port

SiteMinder 를 호스트하는 웹 서버 및 포트를 식별합니다.

SP_ID

파트너 관계에서 정의된 서비스 공급자의 엔터티 ID 를 지정합니다. 엔터티 ID 는 대/소문자를 구분합니다. "관리 UI"에 표시되는 그대로 입력하십시오.

URI_for_binding

ProtocolBinding 요소에 대한 POST 또는 아티팩트 바인딩의 URI 를 식별합니다. 이 URI 는 SAML 2.0 사양에 의해 정의됩니다.

- SAML 2.0 사양으로 지정된 아티팩트 바인딩 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

- SAML 2.0 사양으로 지정된 POST 바인딩 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST

  HTTP-POST 싱글 사인온에 대해서는 이 매개 변수를 설정할 필요가 없습니다.

참고: 요청이 작동하려면 파트너 관계에 바인딩을 사용할 수 있어야 합니다.

target_URL

서비스 공급자에서 페더레이션 리소스 대상의 URL 을 지정합니다.

다음에 주의하십시오.

- 링크에 ProtocolBinding 쿼리를 포함하지 않은 경우 서비스 공급자 속성에 구성된 바인딩 하나를 사용하십시오.
- 아티팩트 및 POST 가 서비스 공급자 속성에서 사용하도록 설정된 경우에는 POST 가 기본값입니다. 따라서 아티팩트 바인딩만 사용하려면 링크에 ProtocolBinding 쿼리 매개 변수를 포함하십시오.

중요! 어설션 소비자 서비스에 대해 인덱싱된 끝점 지원을 구성하면 ProtocolBinding 쿼리 매개 변수의 값이 어설션 소비자 서비스에 대한 바인딩을 무시합니다.

IdP 에서 사용되는 원치 않는 응답 쿼리 매개 변수

IdP 에서 싱글 사인온을 시작하는 원치 않는 응답에는 다음 쿼리 매개 변수가 포함될 수 있습니다.

SPID

(필수) 아이덴티티 공급자가 원치 않는 응답을 보내는 서비스 공급자의 ID 를 지정합니다. 엔터티 ID 는 대/소문자를 구분합니다. "관리 UI"에 표시되는 그대로 입력하십시오.

ProtocolBinding

원치 않는 응답의 ProtocolBinding 요소를 지정합니다. 이 요소는 서비스 공급자에게 어설션 응답을 보내기 위한 프로토콜을 지정합니다. 서비스 공급자가 지정된 프로토콜 바인딩을 지원하도록 구성되어 있지 않으면 요청이 실패합니다.

RelayState

서비스 공급자에서 대상 리소스의 URL 을 나타냅니다. 이 쿼리 매개 변수를 포함하면 IdP 에 사용자를 서비스 공급자의 적절한 리소스로 리디렉션하도록 지시하는 것입니다. 이 쿼리 매개 변수는 싱글 사인온을 구성할 때 대상 URL 을 지정하는 대신 사용할 수 있습니다.

ProtocolBinding 쿼리 매개 변수의 필수 사용

ProtocolBinding 매개 변수는 서비스 공급자 속성에서 아티팩트 및 POST 바인딩이 사용되도록 설정한 *경우에만* 필요합니다. 또한 사용자는 아티팩트 바인딩만 사용하려고 합니다.

- 아티팩트 바인딩에 대한 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

- POST 바인딩에 대한 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST

  HTTP-POST 싱글 사인온에 대해서는 이 매개 변수를 설정할 필요가 없습니다.

참고: 쿼리 매개 변수 HTTP 코딩은 필요하지 않습니다.

ProtocolBinding 쿼리 매개 변수의 선택적 사용

ProtocolBinding 쿼리 매개 변수를 사용하지 *않는* 경우 다음 정보가 적용됩니다.

- 서비스 공급자에 대해 하나의 바인딩만 사용되도록 지정되고 원치 않는 응답에 ProtocolBinding 이 지정되지 않은 경우에는 사용되도록 지정된 바인딩이 사용됩니다.
- 서비스 공급자에 대해 두 바인딩이 모두 사용되도록 지정되었고 원치 않는 응답에 ProtocolBinding 이 지정되지 않은 경우에는 POST 바인딩이 기본값입니다.

예: ProtocolBinding 이 없는 원치 않는 응답

링크가 사용자를 싱글 사인온 서비스로 리디렉션합니다. 이 링크에는 SPID 쿼리 매개 변수가 지정하는 서비스 공급자 아이덴티티가 포함되어 있습니다. ProtocolBinding 쿼리 매개 변수는 없습니다. 사용자가 이 하드 코드된 링크를 클릭하면 싱글 사인온 서비스로 리디렉션됩니다.

http://fedsrv.fedsite.com:82/affwebservices/public/saml2sso?
SPID=http%3A%2F%2Ffedsrv.acme.com%2Fsmidp2for90

예: ProtocolBinding 이 있는 원치 않는 응답

링크가 사용자를 싱글 사인온 서비스로 리디렉션합니다. 이 링크에는 SPID 쿼리 매개 변수가 지정하는 서비스 공급자 아이덴티티가 포함되어 있으며 아티팩트 바인딩이 사용됩니다. 사용자가 이 하드 코드된 링크를 클릭하면 로컬 싱글 사인온 서비스로 리디렉션됩니다.

http://idp-ca:82/affwebservices/public/saml2sso?SPID=
http%3A%2F%2Ffedsrv.acme.com%2Fsmidp2for90&
ProtocolBinding=urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

IdP 에서 ForceAuthn 및 IsPassive 처리

서비스 공급자가 싱글 사인온을 시작하는 경우 서비스 공급자는 ForceAuthn 또는 IsPassive 쿼리 매개 변수를 AuthnRequest 메시지에 포함할 수 있습니다.

서비스 공급자가 AuthnRequest 에 ForceAuthn 또는 IsPassive 를 포함하면 SiteMinder 아이덴티티 공급자는 이 쿼리 매개 변수를 다음과 같이 처리합니다.

ForceAuthn 처리

서비스 공급자가 AuthnRequest 메시지에 ForceAuthn=True 를 포함하면 SiteMinder 아이덴티티 공급자는 사용자에게 자격 증명을 묻습니다. SiteMinder 세션이 종료될 때에도 자격 증명을 묻습니다.

IsPassive 처리

SiteMinder IdP 는 피동 인증을 지원하지 않습니다. 서비스 공급자가 AuthnRequest 에 IsPassive 를 포함하지만 아이덴티티 공급자가 이를 처리할 수 없으면 IdP 는 이 SAML 응답 중 하나를 다시 전송합니다.

- AuthnRequest 메시지에 IsPassive=True 가 있고 세션이 없는 경우 아이덴티티 공급자는 오류 메시지를 반환합니다. SiteMinder 에는 세션이 필요합니다.
- AuthnRequest 메시지에 IsPassive=True 가 있고 세션이 있는 경우에는 아이덴티티 공급자가 어설션을 반환합니다.
- IsPassive 및 ForceAuthn 이 AuthnRequest 메시지에 있고 둘 다 True 로 설정된 경우 SiteMinder 아이덴티티 공급자는 오류를 반환합니다. IsPassive 와 ForceAuthn 은 동시에 사용할 수 없습니다.

SP 에서 시작되는 SSO(SAML 2.0)

SP 에서 시작되는 SSO 의 경우에는 서비스 공급자의 AuthnRequest 서비스에 대한 하드 코딩된 링크가 포함된 HTML 페이지가 서비스 공급자에 있어야 합니다. 링크는 사용자를 인증될 아이덴티티 공급자로 리디렉션하고 AuthnRequest 자체에 무엇이 포함되어 있는지 확인합니다.

이 정보는 아티팩트 또는 POST 바인딩에 적용됩니다.

사용자가 선택하는 하드 코딩된 링크에는 AuthnRequest 서비스에 대한 HTTP GET 요청에서 사용되는 특정 쿼리 매개 변수가 포함되어야 합니다.

참고: 이러한 하드 코드된 링크가 포함된 페이지는 보호되지 않은 영역에 있어야 합니다.

트랜잭션에 대해 아티팩트 또는 프로필 바인딩이 사용되도록 지정하기 위한 링크 구문은 다음과 같습니다.

http://*sp_server:port*/affwebservices/public/saml2authnrequest?ProviderID=*IdP_ID*&ProtocolBinding=*URI_of_binding*&RelayState=*target_URL*

sp_server:port

CA SiteMinder® Federation 을 호스트하는 서비스 공급자의 서버 및 포트를 지정합니다.

IdP_ID

아이덴티티 공급자에게 할당된 아이덴티티를 지정합니다. 엔터티 ID 는 대/소문자를 구분합니다. "관리 UI"에 표시되는 그대로 입력하십시오.

URI_of_binding

ProtocolBinding 요소에 대한 POST 또는 아티팩트 바인딩의 URI 를 식별합니다. 이 URI 는 SAML 2.0 사양에 의해 정의됩니다.

- 아티팩트 바인딩에 대한 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

- POST 바인딩에 대한 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST

  HTTP-POST 싱글 사인온에 대해서는 이 매개 변수를 설정할 필요가 없습니다.

또한 요청이 작동하게 하려면 파트너 관계에 대한 바인딩이 사용되도록 설정하십시오.

target_URL

서비스 공급자에서 페더레이션 대상의 URL 을 지정합니다.

다음 정보에 주의하십시오.

- AuthnRequest 링크에 ProtocolBinding 쿼리 매개 변수를 포함하지 않는 경우 기본 바인딩은 파트너 관계에 대해 정의된 바인딩입니다. 두 바인딩이 모두 파트너 관계에 정의되어 있는 경우에는 AuthnRequest 에서 바인딩이 전달되지 않습니다. 따라서 아이덴티티 공급자의 기본 바인딩이 사용됩니다.
- 아티팩트 및 POST 바인딩이 파트너 관계에 대해 사용되도록 설정되어 있지만 아티팩트 바인딩만 사용하려는 경우에는 링크에 ProtocolBinding 쿼리를 포함하십시오.

SP 에서 사용하는 AuthnRequest 쿼리 매개 변수

SiteMinder SP 가 AuthnRequest 서비스에 대한 링크에 사용할 수 있는 쿼리 매개 변수는 다음과 같습니다.

ProviderID(필수)

AuthnRequest 서비스가 AuthnRequest 메시지를 보내는 아이덴티티 공급자의 엔터티 ID 입니다. 엔터티 ID 는 대/소문자를 구분합니다. "관리 UI"에 표시되는 그대로 입력하십시오.

ProtocolBinding

AuthnRequest 메시지의 ProtocolBinding 요소를 지정합니다. 이 요소는 아이덴티티 공급자의 SAML 응답을 반환하기 위한 프로토콜을 지정합니다. 지정된 아이덴티티 공급자가 지정된 프로토콜 바인딩을 지원하도록 구성되어 있지 않으면 요청이 실패합니다.

AuthnRequest 에서 이 매개 변수를 사용하는 경우에는 AssertionConsumerServiceIndex 매개 변수를 포함할 수 없습니다. 두 매개 변수는 동시에 사용할 수 없습니다.

ForceAuthn

아이덴티티 공급자에게 기존 보안 컨텍스트를 사용하는 대신 사용자를 직접 인증해야 한다고 지시합니다. 아이덴티티 공급자가 CA SiteMinder® Federation 을 사용하면 이 쿼리 매개 변수를 사용하고, 타사 페더레이션 소프트웨어를 사용하면 이 쿼리 매개 변수를 사용하지 마십시오.

- SP 가 AuthnRequest 메시지에 ForceAuthn=True 를 설정하고 특정 사용자에 대한 세션이 존재하는 경우에는 아이덴티티 공급자가 사용자에게 인증을 요청합니다. 사용자가 성공적으로 인증되면 IdP 가 기존 세션의 아이덴티티 정보를 어설션에 넣어 보냅니다. 아이덴티티 공급자는 재인증을 위해 생성하는 세션을 삭제합니다.
- SP 가 AuthnRequest 메시지에 ForceAuthn=True 를 설정한 경우 세션이 없으면 IdP 가 사용자에게 인증을 요청합니다. 사용자가 성공적으로 인증되면 세션이 설정됩니다.

예

http://sp1.demo.com:81/affwebservices/public/saml2authnrequest?ProviderID=idp1.example.com&ForceAuthn=yes

IsPassive

아이덴티티 공급자가 사용자에게 자격 증명을 요구하지 않고, 또는 어떤 방식으로든 사용자와 상호 작용하지 않고 사용자를 로그인하도록 지시합니다. SiteMinder 아이덴티티 공급자는 사용자에게 세션이 없으면 이 쿼리 매개 변수를 인정하지 않습니다. 사용자에게 세션이 없으면 아이덴티티 공급자는 오류를 반환합니다.

AssertionConsumerServiceIndex

어설션 소비자 서비스로 작동하는 끝점의 인덱스를 지정합니다. 인덱스는 아이덴티티 공급자에게 어설션 응답을 보낼 위치를 알려 줍니다.

AuthnRequest 에서 이 매개 변수를 사용하는 경우에는 ProtocolBinding 매개 변수를 포함하지 마십시오. 이 매개 변수와 ProtocolBinding 매개 변수는 함께 사용할 수 없습니다. 어설션 소비자 서비스에는 자체 프로토콜 바인딩이 있으며 이는 ProtocolBinding 매개 변수와 충돌할 수 있습니다.

RelayState

서비스 공급자에서 대상 리소스의 URL 을 나타냅니다. 이 쿼리 매개 변수를 포함하면 서비스 공급자에게 사용자를 보낼 대상을 알려 줍니다. 그렇지 않으면 파트너 관계의 기본 대상이 사용됩니다.

ProtocolBinding 쿼리 매개 변수의 필수 사용

아티팩트 및 POST 바인딩이 파트너 관계에 대해 사용되도록 설정되어 있고 사용자가 아티팩트 바인딩만 사용하고자 하는 경우에는 ProtocolBinding 매개 변수가 필요합니다.

- 아티팩트 바인딩에 대한 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

- POST 바인딩에 대한 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST

  HTTP-POST 싱글 사인온에 대해서는 이 매개 변수를 설정할 필요가 없습니다.

ProtocolBinding 의 선택적 사용

ProtocolBinding 쿼리 매개 변수를 사용하지 *않는* 경우 다음 조건이 적용됩니다.

- 파트너 관계에 대해 하나의 바인딩만 사용되도록 설정되어 있고 ProtocolBinding 쿼리 매개 변수가 지정되지 않은 경우에는 파트너 관계에 사용되도록 설정된 바인딩이 사용됩니다.
- 두 바인딩이 모두 사용되도록 설정된 경우 ProtocolBinding 쿼리 매개 변수를 지정하지 않으면 POST 바인딩이 기본값으로 사용됩니다.

참고: 쿼리 매개 변수를 HTTP-인코딩할 필요는 없습니다.

예: ProtocolBinding 쿼리 매개 변수가 없는 AuthnRequest 링크

이 샘플 링크는 AuthnRequest 서비스로 이동합니다. 이 링크는 ProviderID 쿼리 매개 변수에 있는 아이덴티티 공급자를 지정합니다.

http://ca.sp.com:90/affwebservices/public/saml2authnrequest?
ProviderID=http%3A%2F%2Ffedsrv.acme.com%2Fsmidp2for90

사용자가 서비스 공급자에서 링크를 클릭하면 SiteMinder 가 AuthnRequest 메시지에 대한 요청을 전달합니다.

예: ProtocolBinding 쿼리 매개 변수가 있는 AuthnRequest 링크

http://ca.sp.com:90/affwebservices/public/saml2authnrequest?
ProviderID=http%3A%2F%2Ffedsrv.acme.com%2Fsmidp2for90&
ProtocolBinding=urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

사용자가 서비스 공급자에서 링크를 클릭하면 SiteMinder 가 AuthnRequest 메시지에 대한 요청을 전달합니다.

IP 에서 시작되는 싱글 사인온(WSFED)

사용자가 RP(리소스 파트너)로 이동하기 전에 IP(아이덴티티 공급자)를 방문할 수 있습니다. 사용자가 먼저 아이덴티티 공급자로 이동하면 링크가 HTTP Get 요청을 생성해야 합니다. 하드 코딩된 링크는 IP 의 피동 요청자 서비스를 가리킵니다. 요청에는 RP 공급자 ID 와 선택적으로 기타 매개 변수가 포함됩니다.

링크 구문은 다음과 같습니다.

https://i*p_server:port*/affwebservices/public/wsfedsso?wa=wsignin1.0&wtrealm=*rp_id*

ip_server:port

아이덴티티 파트너에 있는 시스템의 서버 및 포트 번호를 지정합니다. 시스템은 페더레이션 네트워크에 설치된 구성 요소에 따라 웹 에이전트 옵션 팩이나 SPS 페더레이션 게이트웨이를 호스트하고 있습니다.

rp_id

RP 의 ID 입니다. 엔터티 ID 는 대/소문자를 구분합니다. "관리 UI"에 표시되는 그대로 입력하십시오.

RP 에서 시작되는 싱글 사인온(WSFED)

사용자가 RP 에서 싱글 사인온을 시작하는 경우 일반적으로 사용자는 목록에서 IP 를 선택합니다. 사이트 선택 페이지는 보호되지 않은 영역에 있습니다.

사이트 선택 페이지의 링크는 IP 의 피동 요청자 서비스를 가리킵니다. 링크가 선택된 후 RP 는 어설션을 가져오기 위해 사용자를 IP 로 리디렉션합니다.

제 16 장: 사용자 세션에서 로그아웃

싱글 로그아웃 개요(SAML 2.0)

SLO(싱글 로그아웃)가 수행되면 로그아웃을 시작한 브라우저에 대한 모든 사용자의 세션이 동시에 종료됩니다. 모든 세션을 닫으면 인증되지 않은 사용자가 SP 의 리소스에 액세스하지 못하게 됩니다.

싱글 로그아웃이 반드시 사용자의 모든 세션을 종료하는 것은 아닙니다. 예를 들어 브라우저를 두 개 열어 둔 사용자에게는 두 개의 독립적 세션이 있을 수 있습니다. 이 경우 로그아웃을 시작하는 브라우저에 대한 세션만 해당 세션에 대한 모든 페더레이션된 사이트에서 종료됩니다. 다른 브라우저의 세션은 여전히 활성 상태입니다.

싱글 로그아웃 바인딩에 따라 싱글 로그아웃 메시지와 함께 전송되는 내용 및 수신되는 각 메시지를 처리하는 방법이 결정됩니다.

중요! 싱글 로그아웃을 구성하려면 정책 서버 관리 콘솔을 사용하여 세션 저장소가 사용되도록 설정하십시오. 관리 콘솔 사용에 대한 자세한 내용은 *정책 서버 관리 안내서*를 참조하십시오.

싱글 로그아웃 작업에는 다음 두 개의 바인딩을 사용할 수 있습니다.

HTTP-리디렉션

HTTP-리디렉션 바인딩은 브라우저를 사용하여 각 로그아웃 트랜잭션을 수행합니다. 싱글 로그아웃 메시지는 항상 GET 요청입니다. 모든 요청 및 응답에 브라우저가 관여합니다. 브라우저가 관여한다는 것은 HTTP-리디렉션 바인딩은 SOAP 바인딩과 달리 브라우저 세션 데이터를 제공한다는 것을 의미합니다.

HTTP-리디렉션 바인딩의 단점은 메시지의 데이터가 쿼리 문자열에서 전송할 수 있는 데이터로 제한된다는 점입니다. 또한 HTTP-리디렉션 바인딩은 익명 프로세스이므로 시간 만료가 발생할 가능성이 별로 없습니다. 하지만 리디렉션에 실패하면 이로 인해 전체 싱글 로그아웃 체인이 중지됩니다.

SOAP

SOAP 바인딩은 POST 요청을 사용하여 싱글 로그아웃 트랜잭션을 수행합니다. POST 요청을 통해 HTTP-리디렉션 바인딩보다 많은 데이터를 전송할 수 있습니다. 또한 SOAP 를 통해 더 다양한 암호화 방법 및 다른 기능을 사용할 수 있습니다.

SOAP 는 동기식 프로세스입니다. IdP 는 더 많은 제어권을 가지며 하나의 SP 에서 발생하는 문제로 프로세스 전체가 방해되는 상황을 방지할 수 있습니다. SOAP 통신은 백 채널을 통해 수행됩니다. 한 번의 로그아웃 실패로 IdP 가 나머지 SP 에서 로그아웃 시도를 중지할 필요는 없습니다.

SOAP 는 백 채널 연결을 사용하므로 초기 싱글 로그아웃 호출 및 응답 이후에는 브라우저가 개입되지 않습니다. SOAP 바인딩은 로그아웃 프로세스의 일부로 원격 엔터티에서 쿠키를 정리하지 않습니다. 쿠키는 로컬 엔터티에서만 정리됩니다. 쿠키를 삭제해야 하는 경우 HTTP-리디렉션 바인딩을 사용하십시오.

HTTP-리디렉션 및 SOAP 를 사용하여 네트워크에서 싱글 로그아웃 관리

네트워크에는 HTTP-리디렉션 바인딩을 지원하는 사이트와 SOAP 바인딩을 지원하는 사이트가 있을 수 있습니다. IdP 는 여러 바인딩을 관리해야 하지만 SP 는 하나의 로그아웃 요청만 보내거나 받습니다.

다음 단원에서는 혼합 바인딩 환경을 처리하기 위한 구성 지침을 제공합니다.

SiteMinder 가 IdP 에 있을 때의 SLO 구성

SiteMinder 가 IdP 에 있을 때는 파트너 관계에 HTTP 리디렉션 기반 SLO 서비스 URL 과 SOAP 기반 SLO 서비스 URL 이 포함되도록 구성하십시오.

IdP 에서 SiteMinder 는 세션의 각 SP 에 대한 구성을 검사하고 SOAP 를 사용하는 모든 로그아웃을 먼저 처리합니다. SOAP 를 지원하지 않는 SP 에 대한 HTTP-리디렉션 로그아웃이 그 다음에 처리됩니다.

SiteMinder 가 SP 에 있을 때의 SLO 구성

SiteMinder 가 SP 에 있고 SP 가 싱글 로그아웃을 시작하는 경우에는 HTTP-리디렉션 바인딩으로 로그아웃을 시작하는 것이 좋습니다. 사용자 세션에 대한 다른 SP 는 SOAP 를 지원하지 않을 수 있습니다.

HTTP-리디렉션은 브라우저 세션을 사용하여 모든 리디렉션을 처리합니다. 이러한 이유로 HTTP-리디렉션은 HTTP 리디렉션만 지원하는 SP 의 로그아웃을 위해 IdP 에 있어야 하는 필수 데이터를 전송합니다. SP 가 HTTP-리디렉션으로 프로세스를 시작하는 경우 IdP 는 이를 지원하는 모든 SP 에 SOAP 를 사용할 수 있습니다. 나머지 SP 에 대해서는 HTTP-리디렉션 바인딩으로 전환하십시오.

SOAP 바인딩을 사용하여 싱글 로그아웃을 시작하는 경우에는 브라우저 세션 데이터가 존재하지 않습니다.

SP 에서 시작되는 로그아웃이 HTTP-리디렉션이 사용되도록 하려면 SP 의 로컬 서블릿을 가리키는 HTTP-리디렉션 링크를 페이지나 응용 프로그램에 포함하십시오. SiteMinder 의 경우 이 링크는 다음과 같습니다.

http://s*p_host:port*/affwebservices/public/saml2slo

이 포함된 링크는 SiteMinder 가 IdP 의 SLO 서비스로 보내는 SAML <LogoutRequest> 메시지를 생성하도록 만듭니다. 사용자가 로그아웃하면 먼저 SP 에서 로그아웃이 수행된 다음 로그아웃 요청이 IdP 로 전송됩니다. 그러면 IdP 는 사용자 세션에 관여한 다른 모든 SP 에 대해 로그아웃 프로세스를 완료합니다.

SLO 요청 유효 기간에 대한 차이 시간 이해

로그아웃 요청의 유효 기간을 계산할 때는 두 개의 값이 관련됩니다. 이 값은 IssueInstant 값과 NotOnOrAfter 값입니다. SLO 응답에서 싱글 로그아웃 요청은 NotOnOrAfter 값에 도달할 때까지 유효합니다. 싱글 로그아웃 요청이 생성될 때 SiteMinder 의 시스템 시간이 사용됩니다. 그 결과로 얻은 시간은 요청 메시지에 설정되는 IssueInstant 가 됩니다. 로그아웃이 만료되는 시점을 확인하기 위해 SiteMinder 는 현재 시간을 가져와서 여기에 "차이 시간"과 "SLO 유효 기간"을 더합니다. 그 결과로 얻은 시간은 NotOnOrAfter 값이 됩니다.

참고: 시간은 GMT 를 기준으로 합니다.

예를 들어 어설션 당사자 측에서 로그아웃 요청이 1:00 GMT 에 생성된다고 가정합니다. 차이 시간은 30 초이고 SLO 유효 기간은 60 초입니다. 따라서 요청은 1:00 GMT 에서 1:01:30 GMT 까지 유효합니다. IssueInstant 값은 1:00 GMT 이고 싱글 로그아웃 요청 메시지는 90 초 뒤에 더 이상 유효하지 않습니다.

싱글 로그아웃 구성

싱글 로그아웃을 구성하려면 정책 서버 관리 콘솔을 사용하여 세션 저장소가 사용되도록 설정해야 합니다. 관리 콘솔 사용에 대한 자세한 내용은 *정책 서버 관리 안내서*를 참조하십시오. 세션 저장소가 사용되도록 설정되지 않은 경우에는 관리 UI 에 싱글 로그아웃 설정이 표시되지 않습니다.

싱글 로그아웃을 구성할 때는 다음 정보를 참조하십시오.

- 파트너가 HTTP-리디렉션을 사용하여 SAML <LogoutRequest> 메시지를 수신하는 경우 보내는 당사자에 대한 응답은 HTTP-리디렉션 바인딩을 사용해야 합니다.
- 파트너가 SOAP 를 사용하여 SAML <LogoutRequest> 메시지를 수신하는 경우 보내는 당사자에 대한 응답은 SOAP 를 통해 전송되어야 합니다.
- 파트너가 지원하지 않는 바인딩을 통해 SLO 를 수신하는 경우에는 싱글 로그아웃이 실패합니다.
- 싱글 로그아웃 사용자 세션에 HTTP-리디렉션 및 SOAP 바인딩을 사용하는 파트너가 포함된 경우에는 두 바인딩을 모두 지원하도록 SiteMinder 를 구성하십시오. IdP 는 로그아웃을 진행할 때 SOAP 를 사용하여 모든 SP 를 로그아웃한 다음 HTTP-리디렉션 바인딩을 사용하여 모든 SP 를 로그아웃합니다.
- SiteMinder SP 가 싱글 로그아웃을 시작하는 경우에는 SP 가 SOAP 를 지원하더라도 HTTP-리디렉션 바인딩부터 사용하여 시작합니다.

  SOAP 및 HTTP-리디렉션이 지원되는 혼합 환경에서 싱글 로그아웃 관리 (페이지 234)에 대한 지침을 검토하십시오.

다음 단계를 수행하십시오.

참고: SLO 구성 설정은 IdP 와 SP 에서 동일합니다.

1. 파트너 관계 마법사의 SSO 및 SLO 단계부터 시작합니다.
2. SLO 섹션에서 SLO 바인딩을 하나 또는 둘 다 선택합니다.

   SLO 바인딩은 싱글 로그아웃을 가능하게 하며 로컬 엔터티에서 사용 중인 바인딩을 나타냅니다. 또한 SLO 바인딩은 로컬 엔터티가 싱글 로그아웃 요청을 수신할 때 수락하는 바인딩도 나타냅니다.

   SOAP 를 선택하면 SOAP 메시지에서 이름 ID 를 암호화할 수 있습니다. 암호화 옵션은 파트너 관계 마법사의 "서명 및 암호화" 단계에서 설정합니다.

SOAP 를 바인딩으로 선택하면 백 채널에 대한 수신 및 송신 구성이 활성화됩니다. SLO 요청 및 응답이 백 채널을 통해 전송됩니다. 각 로컬 파트너는 원격 파트너에게 인증을 요청하여 백 채널에 보안을 적용할 수 있습니다.

SLO 에 대한 백 채널 설정 (페이지 238)에 대한 더 많은 내용을 확인할 수 있습니다.

3. 추가 SLO 설정을 구성합니다.
   - SLO 확인 URL
   - SLO 유효 기간(초)
   - 릴레이 상태가 SLO 확인 URL 무시

   필드 설명을 보려면 "도움말"을 클릭하십시오.
4. SLO 서비스 URL 에 대한 테이블을 완성합니다. 최소한 하나의 항목이 있어야 합니다. 선택된 원격 엔터티에 대해 정의된 값이 이미 테이블에 입력되어 있어야 합니다.

   SLO 서비스 URL 은 싱글 로그아웃을 시작한 다음 SAML <LogoutRequest> 메시지를 생성하도록 정책 서버를 트리거합니다. 또한, SLO 서비스 URL 은 정책 서버에게 로그아웃 요청 메시지를 보낼 곳을 알려 줍니다.

   지원되는 각 SLO 바인딩에 대해 다음과 같이 SLO 서비스 URL 을 지정합니다.
   - HTTP-Redirect 사용 - HTTP-Redirect 를 바인딩으로 사용하는 하나의 URL 을 선택합니다.
   - SOAP 사용 - SOAP 를 바인딩으로 사용하는 하나의 URL 을 선택합니다.
   - 리디렉션 및 SOAP 사용 - 두 개의 URL(HTTP-리디렉션으로 설정된 URL 한 개와 SOAP 로 설정된 URL 한 개)을 선택합니다.

   참고: "응답 위치 URL" 필드는 선택 사항입니다.

싱글 로그아웃 구성이 완료되었습니다.

싱글 로그아웃에 대한 백 채널 구성

SOAP 바인딩을 사용한 싱글 로그아웃은 로그아웃 요청 및 응답을 백 채널을 통해 전송합니다. 엔터티가 백 채널에 액세스하려면 인증이 필요하도록 설정할 수 있습니다. SSL 은 필수가 아니지만 SSL 을 사용하여 백 채널에 보안을 적용할 수도 있습니다.

SSL 을 사용하여 백 채널에 보안을 적용하는 절차는 다음과 같습니다.

- SSL 이 사용되도록 설정합니다.

  기본 인증에는 SSL 이 필요하지 않지만 SSL 을 통한 기본 인증을 사용할 수 있습니다. 클라이언트 인증서 인증에는 SSL 이 필요합니다.

- 싱글 로그아웃 교환에 대해 들어오는 백 채널과 나가는 백 채널을 구성합니다. 로컬 엔터티는 나가는 채널을 통해 메시지를 보내고 들어오는 채널을 통해 메시지를 받을 수 있어야 합니다.

  참고: 들어오고 나가는 백 채널을 구성할 수 있지만 한 채널은 하나의 구성만 가질 수 있습니다. 동일한 채널을 사용하는 두 서비스는 동일한 백 채널 구성을 사용합니다. 예를 들어 로컬 어설션 당사자의 수신 채널이 HTTP-아티팩트 SSO 와 SOAP 기반 SLO 를 지원할 경우 이 두 서비스는 동일한 백 채널 구성을 사용해야 합니다.

- 원격 엔터티가 보호된 백 채널을 통해 액세스를 얻기 위한 인증 유형을 선택합니다. 인증 방법은 채널별로(나가는 채널 또는 들어오는 채널) 적용됩니다.

  백 채널 인증에 대한 옵션은 다음과 같습니다.

  기본

  기본 인증 체계로 백 채널을 보호합니다.

  참고: 백 채널 연결에 대해 SSL 이 사용되도록 설정하는 경우에도 기본 인증을 선택할 수 있습니다.

  클라이언트 인증서

  X.509 클라이언트 인증서를 사용한 SSL 이 어설션 당사자 백 채널을 보호합니다.

  "클라이언트 인증서"를 인증 방법으로 선택하는 경우 모든 끝점 URL 이 SSL 통신을 사용해야 합니다. 즉, URL 이 **https://**로 시작해야 합니다. 끝점 URL 은 싱글 사인온, 싱글 로그아웃 및 어설션 소비자 서비스와 같은 다양한 SAML 서비스를 서버에서 찾습니다.

인증 없음

신뢰 당사자가 자격 증명을 제공할 필요가 없습니다. 백 채널에 보안이 적용되지 않습니다. 이 옵션을 사용할 때도 SSL 을 활성화할 수 있습니다. 백 채널 트래픽은 암호화되지만 당사자 간에 자격 증명이 교환되지 않습니다.

"인증 없음" 옵션은 테스트 용도로만 사용하고 프로덕션에는 사용하지 마십시오. SiteMinder 가 SSL 사용 장애 조치를 구현하는 프록시 서버 뒤에 있는 경우는 예외입니다. 백 채널을 보호하기 위해 클라이언트 인증서 인증이 사용되는 경우에는 프록시 서버가 인증을 처리합니다. 모든 IdP->SP 파트너 관계가 "인증 없음"을 인증 유형으로 사용할 수 있습니다.

중요! 들어오는 백 채널에 대한 인증 방법은 파트너 관계에서 다른 측의 나가는 백 채널과 일치해야 합니다. 인증 방법의 선택에 대한 동의는 대역 외 통신에서 처리됩니다.

싱글 로그아웃에 대한 백 채널에 보안을 적용하려면

1. 파트너 관계 마법사의 SSO 및 SLO 단계에 있는 "백 채널" 섹션부터 시작합니다.
2. SLO 섹션에서 SOAP 를 선택합니다. "인증 방법" 필드가 활성화됩니다.
3. 들어오는 백 채널 및 나가는 백 채널에 대한 인증 방법의 유형을 선택합니다. "기본" 및 "클라이언트 인증서" 방법에 대해 구성할 추가 필드가 표시됩니다.

   "인증 없음"을 인증 방법으로 선택하는 경우에는 추가적인 단계가 필요 없습니다.
4. 선택하는 인증 방법에 따라 구성해야 할 몇 개의 추가적인 필드가 표시됩니다.

모든 필수 필드에 값을 입력하면 백 채널 구성이 완료됩니다.

사인아웃 개요(WS-페더레이션)

사인아웃은 사인아웃을 시작한 브라우저에 대해 모든 사용자 세션을 동시에 종료하는 것입니다. 모든 사용자 세션을 닫으면 권한 없는 사용자가 리소스 파트너의 리소스에 액세스하지 못하게 됩니다.

사인아웃이 반드시 사용자의 모든 세션을 종료하는 것은 아닙니다. 예를 들어 브라우저를 두 개 열어 둔 사용자에게는 두 개의 독립적 세션이 있을 수 있습니다. 이 경우 사인아웃을 시작하는 브라우저에 대한 세션만 해당 세션에 대한 모든 페더레이션된 사이트에서 종료됩니다. 다른 브라우저의 세션은 여전히 활성 상태입니다.

정책 서버는 signoutconfirmurl.jsp 를 사용하여 사인아웃을 수행합니다. 이 페이지는 아이덴티티 공급자 시스템에 있습니다. 아이덴티티 공급자 파트너는 사용자 대신 사인아웃 요청을 시작합니다. JSP 는 사용자가 특정 브라우저 세션 중에 사인온한 각 사이트에 사인아웃 요청을 보냅니다. 그러면 사용자가 사인아웃됩니다.

사용자는 아이덴티티 공급자에서만 사인아웃 요청을 시작할 수 있습니다. 적절한 서블릿을 가리키는 링크를 클릭하면 요청이 트리거됩니다. 아이덴티티 공급자 사이트에서 사인아웃 확인 페이지는 보호되지 않는 리소스여야 합니다.

참고: 정책 서버는 사인아웃에 대해 WS-페더레이션 피동 요청 프로필만 지원합니다.

WSFED 사인아웃이 사용되도록 설정

사인아웃을 구성하려면 다음 요구 사항을 충족해야 합니다.

- 아이덴티티 공급자에서 사인아웃이 사용되도록 설정하려면 정책 서버 관리 콘솔을 사용하여 세션 저장소가 사용되도록 설정합니다.

  세션 저장소에 대한 자세한 내용은 *정책 서버 관리 안내서*를 참조하십시오.

- 싱글 사인온은 유효한 SiteMinder 영구 세션이 필요합니다. 이 세션은 싱글 사인온 중에 구성됩니다. 리소스 파트너에서 인증 URL 을 포함하여 보호된 리소스가 포함된 영역에 대해 영구 세션을 구성하십시오.

  영역에 대한 자세한 내용은 *정책 서버 구성 안내서*를 참조하십시오.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. 수정할 WS-Federation 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "싱글 사인온 및 사인아웃" 단계로 이동합니다.
4. "사인아웃" 섹션에서 다음 필드를 설정합니다.
   - 사인아웃 사용
   - 사인아웃 확인 URL(IP 만 해당)
   - Sign-Out URL

   각 URL 의 입력은 https:// 또는 http://로 시작해야 합니다.
5. "확인" 단계로 이동하고 "마침"을 클릭하여 변경 내용을 저장합니다.

사인아웃이 구성되었습니다.

SP 에서 로컬 로그아웃(SAML 2.0)

SP 역할을 하는 SiteMinder 는 독립 실행형 응용 프로그램에 대한 로컬 로그아웃을 지원합니다. 로컬 로그아웃의 경우 사용자가 로컬 SP 측 응용 프로그램에서 로그아웃할 수 있습니다. SP 의 세션이 제거되지만 IdP 또는 다른 SP 와의 통신이 연관되지 않습니다. IdP 및 다른 SP 의 세션은 활성 상태로 유지됩니다.

SP 의 응용 프로그램에 로그아웃 링크를 포함하면 SP 는 로컬 싱글 로그아웃 서비스에 로그아웃 요청을 전송합니다. SP 는 요청을 수신하면 사용자를 로그아웃시킵니다. SP 의 응용 프로그램은 로그아웃에 성공했다는 확인 메시지를 보내는 역할을 담당합니다.

SiteMinder 는 **localLogout** 이라는 쿼리 매개 변수를 사용하여 로컬 로그아웃을 제공합니다. 이 매개 변수를 사용하기 위해 응용 프로그램에 다음과 같은 내용의 페이지가 있을 수 있습니다.

```
demoapp 을 사용하여 등록을 완료했습니다.
세션을 안전하게 종료하려면 LOGOUT(로그아웃)을 선택하십시오.
```

다음 샘플 문자열은 LOGOUT(로그아웃) 버튼에 대한 링크를 나타냅니다.

```
<http://sp1server.demo.com:8080/affwebservices/public/saml2slo?LocalLogout=true
```

제 17 장: 인증 컨텍스트 처리(SAML 2.0)

*인증 컨텍스트*는 아이덴티티 공급자에서 사용자가 인증되는 방법을 나타냅니다. 아이덴티티 공급자는 서비스 공급자의 요청에 따라, 또는 아이덴티티 공급자의 구성에 따라 싱글 사인온 어설션에 인증 컨텍스트를 포함합니다. 서비스 공급자는 리소스에 대한 액세스를 허용하기 전에 어설션에 신뢰 수준을 설정하기 위해 인증 프로세스에 대한 정보를 요청할 수 있습니다.

인증 컨텍스트 요청

인증 컨텍스트를 요청하려면 SiteMinder 서비스 공급자가 아이덴티티 공급자에 대한 인증 요청에 <RequestedAuthnContext> 요소를 포함시켜야 합니다. 서비스 공급자는 SP->IdP 파트너 관계의 구성 설정에 따라 요청에 이 요청을 포함시킵니다.

인증 컨텍스트 가져오기

SiteMinder 아이덴티티 공급자는 다음 두 가지 방법 중 *하나*를 사용하여 인증 컨텍스트를 가져옵니다.

- IdP->SP 파트너 관계 구성에서 정적 AuthnContext URI 를 지정합니다.

  페더레이션된 파트너가 AuthnContext 요청을 지원하지 않는 SiteMinder 서비스 공급자인 경우 관리 UI 에 URI 를 직접 입력하십시오.

- AuthnContext URI 는 구성된 인증 컨텍스트 템플릿을 사용하여 동적으로 결정됩니다.

  정책 서버는 인증 컨텍스트 URI 를 정책 서버에서 정의된 인증 수준에 매핑합니다. 이 인증 수준은 연결된 사용자 세션에 대한 인증 컨텍스트의 강도를 나타냅니다. 이 수준에서는 아이덴티티 공급자의 사용자 세션에서 인증 컨텍스트를 추출할 수 있습니다.

  아이덴티티 공급자는 요청을 수신하면 <RequestedAuthnContext> 요소의 값을 인증 컨텍스트와 비교합니다. 비교는 서비스 공급자의 요청에 있는 비교 값을 기반으로 합니다. 비교가 성공적인 경우 아이덴티티 공급자는 서비스 공급자에 반환하는 어설션에 인증 컨텍스트를 포함합니다. 서비스 공급자에서 유효성 검사가 구성된 경우 서비스 공급자는 들어오는 인증 컨텍스트를 요청한 값과 비교하여 유효성을 검사합니다.

IdP 에서 시작되는 SSO 에 대한 인증 컨텍스트 처리

싱글 사인온이 IdP 에서 시작되는 경우 인증 컨텍스트 처리의 단계는 다음과 같습니다.

1. 사용자 요청이 IdP 에서 싱글 사인온을 트리거합니다.
2. 사용자가 인증되고 사용자 세션이 생성됩니다. 인증 체계를 사용하여 구성된 보호 수준이 세션과 연결됩니다.
3. IdP 의 인증 컨텍스트 구성에 따라 다음과 같은 상황 중 *하나*가 발생할 수 있습니다.
   - 자동 검색이 수행됩니다.

     구성된 인증 컨텍스트 템플릿에 기반하여 AuthnContext 클래스가 세션의 보호 수준에 매핑됩니다.
   - 미리 정의된 인증 클래스가 사용됩니다.

     지정한 하드 코딩된 URI 가 어설션에 추가됩니다.
4. IdP 가 어설션을 생성하고 인증 컨텍스트를 여기에 추가합니다. 그런 다음 어설션이 SP 에 전송됩니다.
5. SP 에서 어설션의 인증 컨텍스트 클래스와 SP 의 구성된 인증 클래스 간에 또 다른 비교가 이루어집니다. 이 비교가 성공하면 인증 트랜잭션이 완료됩니다.

SP 에서 시작되는 SSO 에 대한 인증 컨텍스트 처리

싱글 사인온이 SP 에서 시작되는 경우 인증 컨텍스트 처리의 단계는 다음과 같습니다.

1. SP 가 <RequestedAuthnContext> 요소 및 비교 연산자와 함께 인증 요청을 보냅니다. SP-> IdP 파트너 관계 구성의 설정에 따라 요소가 포함됩니다.
2. IdP 가 요청을 받으면 사용자를 인증하고 사용자 세션이 생성됩니다. 인증 체계의 보호 수준이 세션과 연결됩니다.

3. IdP 의 인증 컨텍스트 구성에 따라 다음과 같은 상황 중 *하나*가 발생할 수 있습니다.
   - 자동 검색이 수행됩니다.

     구성된 인증 컨텍스트 템플릿에 기반하여 AuthnContext 클래스가 세션의 보호 수준에 매핑됩니다.
   - 미리 정의된 인증 클래스가 사용됩니다.

     지정한 하드 코딩된 URI 가 어설션에 추가됩니다.
4. IdP 가 AuthnContext 를 사용자 세션의 인증 클래스와 비교합니다. 이 비교는 요청과 함께 전송된 비교 연산자를 기반으로 합니다. 각 비교 연산자가 처리에 미치는 영향의 예를 보려면 이 절차 뒤에 나오는 표를 참조하십시오.

   SP 가 요청에 인증 컨텍스트 URI 를 여러 개 포함하는 경우 각각의 클래스가 일대일로 순차적으로 세션의 컨텍스트와 비교됩니다. 첫 번째 비교가 성공할 경우 IdP 가 세션 인증 컨텍스트를 어설션에 추가합니다.
5. 비교가 성공하면 SP 에 전송되는 어설션에 인증 컨텍스트가 추가됩니다.

   비교가 실패하면 트랜잭션이 종료되고 "noauthncontext" 상태 응답이 반환됩니다.
6. SP 에서 어설션의 인증 컨텍스트와 SP 의 구성된 인증 클래스 간에 두 번째 비교가 이루어집니다. 이 비교가 성공하면 인증 트랜잭션이 완료됩니다.

다음 표에서는 인증 컨텍스트 요청에서 전송된 비교 특성을 기반으로 인증 컨텍스트가 처리되는 방식의 예를 보여 줍니다.

SP 에서 요청된 인증 컨텍스트	비교 특성 값	IdP 에서 구성된 인증 컨텍스트	상태 응답
Password	exact	InternetProtocol	NoAuthnContext
Password	minimum	InternetProtocol	NoAuthnContext
Password	maximum	InternetProtocol	NoAuthnContext
InternetProtocol	exact	InternetProtocol	Success
InternetProtocol	minimum	InternetProtocol	Success

SP 에서 요청된 인증 컨텍스트	비교 특성 값	IdP 에서 구성된 인증 컨텍스트	상태 응답
InternetProtocol	maximum	InternetProtocol	Success
InternetProtocol	maximum	Password	NoAuthnContext
InternetProtocol	maximum	Password	Success

인증 컨텍스트 템플릿 개요

인증 컨텍스트 템플릿은 파트너가 지원하는 특정 SAML 2.0 AuthnContext URI 를 정의합니다. 각 URI 는 특정 컨텍스트 클래스에 보호 수준이 할당되어 있음을 식별하고 그 후에 이 보호 수준이 강도 수준에 매핑됩니다.

파트너 관계 단위로 템플릿을 선택할 수 있으며 여러 파트너 관계가 하나의 템플릿을 사용할 수 있습니다.

템플릿은 각 파트너에서 다음과 같은 고유 기능이 있습니다.

IdP 에서

인증 컨텍스트 템플릿은 SP 요청에서 인증 컨텍스트를 자동으로 감지하도록 IdP 가 구성된 경우 IdP 에서 필요합니다.

템플릿은 URI 를 사용자 세션과 관련된 보호 수준에 매핑합니다. 보호 수준은 1 부터 1000 까지 정책 서버에서 인증 체계의 강도를 나타내며 1000 이 가장 강력한 강도입니다. 관리자는 사용자를 인증하고 사용자 세션을 설정하는 인증 체계를 구성할 때 보호 수준을 할당합니다.

IdP 는 먼저 템플릿을 사용하여 사용자 세션의 강도를 결정합니다. 그런 다음 템플릿을 사용하여 SP 인증 요청에서 URI 의 강도를 결정합니다. 그런 다음 이러한 강도 수준이 비교됩니다.

SP 에서

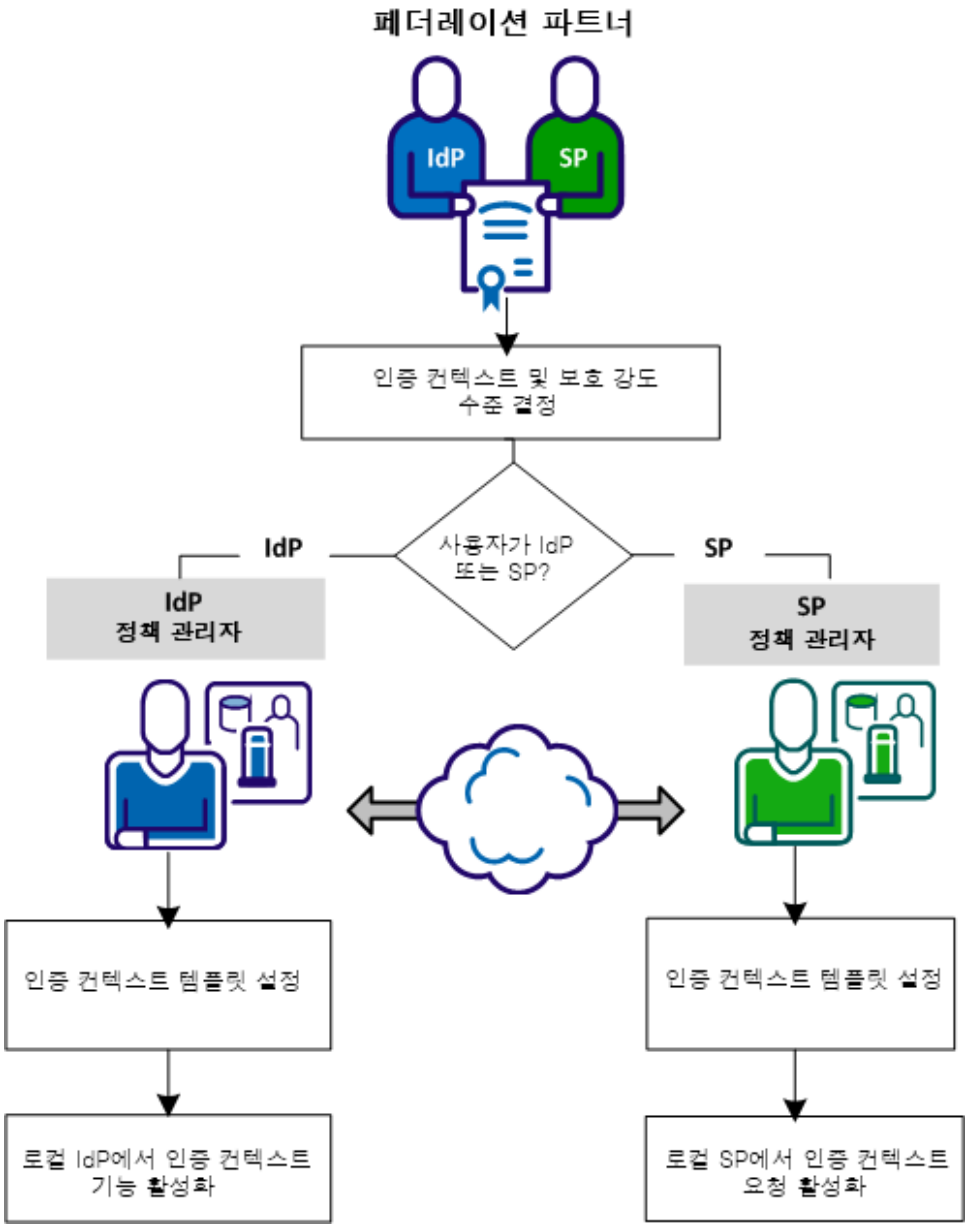
인증 요청을 통해 전달되는 인증 컨텍스트를 생성하려면 SP 의 인증 컨텍스트 템플릿이 필요합니다. SP 는 요청을 생성한 후 IdP 에 전송합니다. 수신한 어설션이 요청된 인증 컨텍스트를 충족하는지 SP 에서 유효성을 검사하는 데도 템플릿이 필요합니다.

구성을 진행하기 전에 최소한 다음과 같은 지식이 있는지 확인하십시오.

- 인증 컨텍스트 처리와 관련한 SAML 2.0 표준
- 페더레이션 구성 개체
- 관리 UI 에 액세스하여 사용하는 방법

인증 컨텍스트 템플릿 구성

-다음 그림에서는 각 파트너의 구성 프로세스를 보여 줍니다. SiteMinder 페더레이션은 각 사이트에 설치되어 있지 않아도 됩니다.



인증 컨텍스트 처리를 구성하려면 다음 단계를 완료하십시오.

1. 파트너와 상의하여 인증 컨텍스트 및 강도 수준을 결정합니다.
2. 인증 컨텍스트 템플릿을 설정합니다.
3. 사이트에서 다음 태스크를 완료합니다.
   - 로컬 IdP 파트너 관계에서 인증 컨텍스트 처리가 사용되도록 설정합니다.
   - 로컬 SP 파트너 관계에서 인증 컨텍스트 요청이 사용되도록 설정합니다.

파트너와 상의하여 인증 컨텍스트 및 강도 수준 결정

요청된 리소스에 대한 액세스를 허용하기 전에 SP 가 특정 인증 컨텍스트 및 강도 수준을 요구할 수 있습니다.. SP 에서 리소스의 민감도에 따라 SP 는 IdP 로부터 받는 어설션을 신뢰해야 합니다.

IdP 와 SP 의 관리자들은 지원되는 인증 컨텍스트와 각 인증 컨텍스트 URI 의 상대적인 강도에 대한 지침을 마련해야 합니다. IdP 에서 URI 의 순서 및 관련된 상대적 강도 수준은 IdP 가 SP 에 응답하는 방식에 영향을 줍니다.

예를 들어, SP 는 X.509 인증서에 대한 인증 컨텍스트 및 정확한 비교 값을 요구합니다. IdP 는 적절한 강도 수준에서 요청하는 사용자를 인증하고 인증 컨텍스트의 평가 중에 비교 값을 충족시켜야 합니다.

인증 컨텍스트 템플릿 설정

인증 컨텍스트 처리를 구현하는 데 필요한 인증 컨텍스트 템플릿을 설정하십시오. 이 절차는 아이덴티티 공급자와 서비스 공급자에서 동일합니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "페더레이션", "파트너 관계 페더레이션", "인증 컨텍스트 템플릿"을 선택합니다.

   "인증 컨텍스트 템플릿 보기" 창이 열립니다.
3. "템플릿 만들기"를 선택합니다.

   첫 번째 단계의 템플릿 마법사가 열립니다.

4. 템플릿의 이름을 입력합니다.
5. 다음 작업 중 *하나*를 수행하십시오.
   - URI 를 수동으로 입력하고 "URI 추가"를 클릭합니다.
   - "기본 URI 로드"를 클릭하고 사전 정의된 목록에서 URI 를 선택합니다. URI 를 "사용 가능한 URI"에서 "선택한 URI" 목록으로 이동합니다.
6. 선택한 URI 를 강도 수준별로 정렬합니다. 강도 수준은 내림차순이며, 가장 강한 URI 가 맨 위쪽에 있고 가장 약한 URI 가 맨 아래쪽에 있습니다.
7. "다음"을 클릭합니다.
8. (선택 사항) 같은 강도 수준으로 지정해야 하는 URI 를 연속으로 배치하여 그룹화합니다. "Change Grouping"(그룹화 변경) 화살표를 사용하여 URI 를 그룹 내부로 또는 외부로 이동합니다.
9. "보호 수준 사용"을 클릭합니다.

   보호 수준을 인증 체계에서 URI 로 매핑합니다. 보호 수준은 1 부터 1000 까지 인증 체계의 강도를 나타내며 1000 이 가장 강력한 강도입니다. 개별 URI 는 고유한 보호 수준을 가질 수 있지만 URI 를 그룹화하면 해당 그룹 내의 URI 가 동일한 강도 수준을 공유합니다.

   보호 수준을 할당할 때 다음과 같은 내용을 고려하십시오.

   - 보호 수준은 내림차순으로 할당하십시오. 가장 강력한 컨텍스트를 맨 위에 나열하고 가장 약한 컨텍스트를 맨 아래에 나열하십시오.
   - 최대 보호 수준을 수정할 수 있으며 그렇게 하면 관리 UI 에서 최소값을 계산합니다. 관리 UI 는 수준 범위에 빈 간격이 없는지 확인하여 각 보호 수준마다 연결된 URI 가 있도록 합니다.

   보호 수준 할당에 대한 자세한 내용을 참조하십시오.
10. "마침"을 선택하여 구성을 완료합니다.

템플릿이 완료되었습니다.

컨텍스트 템플릿에 대한 보호 수준 할당

보호 수준은 인증 강도를 나타냅니다. 보호 수준을 선택된 각 인증 컨텍스트 URI 에 할당하십시오. 목록의 각 URI 에 대해 최대 수준을 지정하십시오. 최소 보호 수준은 목록에 나오는 URI 의 최대 수준을 기반으로 자동으로 결정됩니다. 이 범위는 보호 수준을 반영합니다.

보호 수준 할당은 구성된 정책 서버 인증 체계의 보호 수준을 반영해야 합니다. 예를 들어, 정책 서버는 보호 수준 20 의 X.509 인증 체계를 가질 수 있습니다. 템플릿이 지정하는 보호 수준 범위는 20 을 포함해야 합니다. 마지막으로, 정책 서버는 이 보호 수준을 기준으로 URI 강도 수준을 생성합니다.

예

정책 서버에 설정된 인증 체계	보호 수준
urn:oasis:names:tc:SAML:2.0:ac:classes:X509	20
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorContract	15
urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol	10
urn:oasis:names:tc:SAML:2.0:ac:classes:Password	5

각 URI 에 대해 정책 서버는 보호 수준을 URI 강도 수준에 매핑합니다. 이러한 범위는 인증 체계의 보호 수준을 포함합니다. 예를 들면 다음과 같습니다.

- X509 체계는 16 에서 1000 까지의 보호 수준 포함
- MobileTwoFactorContract 는 11 에서 15 까지의 보호 수준 포함
- Internet Protocol 은 6 에서 10 까지의 보호 수준 포함
- Password 는 1 에서 5 까지의 보호 수준 포함

URI	보호 수준 최대	URI 강도
urn:oasis:names:tc:SAML:2.0:ac:classes:X509	1000	4
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorContract	15	3

URI	보호 수준 최대	URI 강도
urn:oasis:names:tc:SAML:2.0:ac:classes:X509	1000	4
urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol	10	2
urn:oasis:names:tc:SAML:2.0:ac:classes:Password	5	1

몇 개의 URI 를 그룹화하면 그룹화를 통해 서로 다른 보호 수준의 URI 가 동일한 URI 강도를 갖게 됩니다. 이 강도는 URI 가 동등하게 간주됨을 의미합니다.

다음 수정된 표는 X.509 URI 및 MobileTwoFactorContract URI 의 그룹을 나타냅니다.

URI	보호 수준 최대	URI 강도
urn:oasis:names:tc:SAML:2.0:ac:classes:X509	1000	3
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorContract	800	3
urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol	700	2
urn:oasis:names:tc:SAML:2.0:ac:classes:Password	200	1

강도 수준의 범위는 목록에 있는 총 그룹 수를 반영합니다. 예를 들어 세 개의 그룹이 있는 경우 강도 수준의 범위는 1 부터 총 그룹 수인 3 까지입니다.

로컬 IdP 파트너 관계에서 인증 컨텍스트 처리 활성화

IdP 로 기능하는 정책 서버는 다음 두 가지 방법으로 어설션을 위한 인증 컨텍스트를 가져올 수 있습니다.

- 미리 정의된 인증 클래스를 사용합니다.

  인증 클래스에 대한 URI 를 지정하고 SP 의 컨텍스트 요청을 무시합니다. 하드 코딩된 항목은 IdP 에서 시작되는 싱글 사인온에 대한 기본 인증 컨텍스트의 역할을 할 수 있습니다.

- 인증 클래스를 자동으로 감지합니다.

  정책 서버는 인증 컨텍스트 템플릿을 사용하여 사용자 세션 인증 컨텍스트를 자동으로 감지합니다.

  IdP 는 SP 의 인증 요청에 <RequestedAuthnContext> 요소가 포함되지 않은 경우에도 템플릿을 사용합니다. 요소가 있으면 IdP 에 의한 추가적인 평가가 트리거되며 어설션에 넣을 수 있는 항목의 선택 범위가 제한됩니다.

  인증 컨텍스트 처리 (페이지 244)의 흐름에 대한 자세한 정보를 참조할 수 있습니다.

다음 단계를 수행하십시오.

1. IdP->SP 파트너 관계 마법사의 SSO 및 SLO 단계로 이동합니다.
2. "인증" 섹션에서 인증 컨텍스트를 가져오는 방법을 지정합니다. 미리 정의된 클래스 또는 자동으로 감지된 클래스를 인증 컨텍스트 템플릿과 함께 사용합니다.
3. 이전 단계에서 선택한 방법의 단계를 수행합니다.
   - 미리 정의된 클래스를 어설션에 포함하려면 "인증 클래스" 풀다운 메뉴에서 URI 를 선택합니다.
   - 세션 컨텍스트와 템플릿에서 클래스를 포함하려면 "인증 컨텍스트 템플릿" 필드에서 템플릿을 선택하거나 "템플릿 만들기"를 클릭합니다.
4. (선택 사항) 인증 컨텍스트를 가져오는 방법에 따라 "RequestedAuthnContext 무시" 확인란을 선택할 수도 있습니다.

다음 표에서는 "AuthnContext 구성" 및 "RequestedAuthnContext 무시" 설정이 함께 작동하는 방식을 보여 줍니다.

AuthnContext 구성	RequestedAuthnContext 무시	SP 에서 AuthnContext 요청	결과
미리 정의된 클래스	선택됨	예	IdP 는 <RequestedAuthnContext>를 무시하고 어설션에 정의된 값을 사용합니다.
미리 정의된 클래스	선택됨	아니요	IdP 는 기본적으로 어설션에 정의된 값을 반환합니다.

AuthnContext 구성	RequestedAuthnContext 무시	SP 에서 AuthnContext 요청	결과
미리 정의된 클래스	선택되지 않음	예	IdP 가 인증 컨텍스트 요청을 처리하도록 구성되지 않았기 때문에 트랜잭션이 실패합니다. IdP 가 SP 에 오류 메시지를 반환합니다.
미리 정의된 클래스	선택되지 않음	아니요	IdP 는 기본적으로 어설션에 정의된 클래스 값을 반환합니다.
클래스 자동 감지	선택됨	예	IdP 는 인증 체계의 보호 수준을 인증 컨텍스트 템플릿과 비교하고 일치하는 인증 URI 를 어설션에 반환합니다. IdP 는 SP 요청의 값을 무시합니다.
클래스 자동 감지	선택됨	아니요	IdP 는 인증 체계의 보호 수준을 인증 컨텍스트 템플릿과 비교하고 일치하는 인증 URI 를 어설션에 반환합니다. IdP 는 SP 요청의 값을 무시합니다.
클래스 자동 감지	선택되지 않음	예	IdP 는 보호 수준을 SP 가 보내는 인증 컨텍스트 클래스와 비교합니다. IdP 는 인증 컨텍스트 템플릿을 사용하여 IdP 가 어설션에 넣는 인증 URI 를 결정합니다.
클래스 자동 감지	선택되지 않음	아니요	IdP 는 인증 체계의 보호 수준을 인증 컨텍스트 템플릿과 비교하고 일치하는 인증 URI 를 어설션에 반환합니다.

로컬 SP 파트너 관계에서 인증 컨텍스트 요청이 사용되도록 설정

인증 컨텍스트는 어설션 인증 문의 일부이며 이는 사용자가 IdP 에서 인증되는 방법을 나타냅니다. SP 는 리소스에 대한 액세스 권한을 부여하기 전에 어설션에 신뢰 수준을 설정하기 위해 인증 프로세스에 대한 정보를 요청할 수 있습니다.

인증 컨텍스트 URI 는 <AuthnContext> 요소 안의 <AuthnContextClassRef> 요소의 값입니다. 각 URI 는 SP 가 IdP 에서 어설션으로 반환하기를 원하는 컨텍스트 클래스를 식별합니다.

SP 의 인증 컨텍스트 템플릿은 다음 정보를 정의합니다.

- SP 가 IdP 에서 수신하고자 하는 URI. 나가는 요청의 경우 템플릿의 URI 는 요청된 리소스에 대한 액세스를 허용하기 전에 SP 가 허용하는 인증 컨텍스트를 나타냅니다.
- 요청의 URI 를 IdP 에 정의된 URI 에 비교하는 방법
- SP 가 URI 를 사용하는 방법. SP 는 나가는 인증 요청에 URI 를 포함할 수 있습니다. 또한 SP 는 들어오는 어설션 응답에서 URI 의 유효성을 검사할 수도 있습니다. URI 사용을 두 기능 모두에 대해 구성할 수 있습니다.

파트너 관계 단위로 템플릿을 선택할 수 있으며 여러 파트너 관계가 하나의 템플릿을 사용할 수 있습니다.

인증 컨텍스트 요청이 사용되도록 설정하거나 SP 파트너 관계를 구성할 때 인증 컨텍스트 템플릿을 구성하십시오.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. 편집할 SP->IdP 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "AuthnContext 구성" 단계로 이동합니다.

   구성 대화 상자가 열립니다.
4. "인증 컨텍스트 처리 사용" 확인란을 선택합니다.

5. 대화 상자의 필드를 입력합니다. "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

   다음 정보에 주의하십시오.

   - 인증 컨텍스트 템플릿이 없는 경우 템플릿 만들기를 선택합니다.
   - "비교" 필드에서는 SP 인증 요청의 URI 를 아이덴티티 공급자에 구성된 URI 와 비교하는 방법을 보여 줍니다.

     "도움말"에 각 비교 연산자에 대한 자세한 설명이 있습니다.
   - "사용 가능한 URI" 목록에서 URI 를 선택하면 사용 가능한 URI 는 선택된 템플릿에 대해 구성된 URI 를 반영합니다. 미리 정의된 템플릿이 없는 경우 "템플릿 만들기"를 클릭하여 구성합니다.

인증 컨텍스트 요청은 아이덴티티 공급자로 전송되는 인증 요청에 포함됩니다.

제 18 장: 페더레이션 메시지 서명 및 암호화

이 섹션은 다음 항목을 포함하고 있습니다.

페더레이션에 대한 키 및 인증서 관리

어설션에 보안을 적용하고 어설션 내의 데이터를 암호화하는 것은 파트너 관계 구성의 중요한 부분입니다. 페더레이션 환경에서는 키/인증서 쌍 및 독립 실행형 인증서가 다음과 같은 여러 기능을 수행합니다.

- 어설션의 서명/확인(3 개 프로필 모두)
- 인증 요청 서명/확인(SAML 2.0 만 해당)
- 싱글 로그아웃 요청 및 응답 서명/확인(SAML 2.0)
- HTTP-아티팩트 SSO 에 대한 채널 요청 및 응답 서명(SAML 1.1 및 2.0)
- 전체 어설션 또는 어설션의 일부 암호화/암호 해독(SAML 2.0)
- 아티팩트 SSO(Single Sign–On)를 위한 백 채널 전반의 클라이언트 자격 증명(SAML 1.1 및 2.0)

*정책 서버 구성 안내서*에는 키 및 인증서 관리에 대한 정보와 지침이 포함되어 있습니다.

SSL 서버 인증서를 사용하여 다음 태스크를 수행할 수 있습니다.

- SSL 연결을 통한 페더레이션 트래픽 관리
- 백 채널을 통한 아티팩트 싱글 사인온의 보안 통신

SiteMinder 웹 에이전트가 설치된 웹 서버에서 SSL 을 사용하도록 설정하는 지침을 참조하십시오.

참고: SSL 이 사용되도록 설정하는 경우 "기준 URL" 매개 변수를 포함한 모든 서비스의 모든 URL 이 영향을 받습니다. 즉, 모든 서비스 URL 이 https://로 시작해야 합니다.

SAML 2.0 서명 알고리즘

SAML 2.0 의 경우 서명 태스크에 대한 서명 알고리즘을 선택할 수 있습니다. 알고리즘을 선택할 수 있으므로 다음과 같은 사용 사례가 지원됩니다.

- IdP 가 RSAwithSHA1 을 사용하는 어설션, 응답 및 SLO-SOAP 메시지 또는 RSAwithSHA256 알고리즘에 서명하는 IdP-->SP 파트너 관계
- SP 가 RSAwithSHA1 을 사용하는 인증 요청과 SLO-SOAP 메시지 또는 RSAwithSHA256 알고리즘에 서명하는 SP-->IdP 파트너 관계

서명 확인은 서명된 문서에서 사용 중인 알고리즘을 자동으로 감지하고 이를 확인합니다. 따라서 서명 확인을 위한 구성은 필요하지 않습니다.

SAML 1.1 생산자 및 WSFED IP 에서 서명 구성

"서명" 단계에서는 정책 서버가 개인 키와 인증서를 사용하여 SAML 어설션 또는 WS-페더레이션 토큰 응답에 서명하는 방식을 정의할 수 있습니다. SAML 1.1 의 경우 어설션 응답 대신 어설션에만 서명하도록 선택할 수 있습니다.

SAML 1.1 과 WS-페더레이션은 암호화를 지원하지 않습니다.

인증서 데이터 저장소에 여러 개의 개인 키와 인증서가 있을 수 있습니다. 여러 개의 페더레이션된 파트너가 있는 경우 각 파트너마다 다른 키 쌍을 사용할 수 있습니다.

참고: 시스템이 FIPS_COMPAT 또는 FIPS_MIGRATE 모드에서 작동하는 경우 풀다운 목록에서 모든 인증서와 키 항목을 사용할 수 있습니다. 시스템이 FIPS 전용 모드에서 작동하는 경우에는 FIPS 승인 인증서 및 키 항목만 사용할 수 있습니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. 수정하려는 어설션 당사자-신뢰 당사자 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "서명" 단계로 이동합니다.
4. "서명" 섹션에 있는 "서명 개인 키 별칭" 필드의 풀다운 목록에서 별칭을 선택합니다.

   인증서 데이터 저장소에 개인 키가 없는 경우 "가져오기"를 클릭하여 키를 가져옵니다. 또는 "생성"을 클릭하여 인증서 요청을 생성합니다.

   이 필드에 데이터를 입력하면 어설션 당사자가 어설션 및 응답에 서명하기 위해 사용하는 개인 키가 지정됩니다.
5. (SAML 1.1 만 해당) "아티팩트" 및 "Post" 서명 옵션에서는 서명을 원하는 특정 구성 요소(어설션, 응답)를 선택합니다.

테스트 환경에서 SiteMinder 를 사용 중인 경우 테스트를 단순화하기 위해 서명 처리를 중지할 수 있습니다. "서명 처리 사용 안 함" 확인란을 클릭하십시오.

서명 구성이 완료되었습니다.

SAML 1.1 소비자 및 WSFED RP 에서의 서명 확인

"서명" 단계에서는 정책 서버가 개인 키와 인증서를 사용하여 SAML 어설션 또는 WS-페더레이션 토큰 응답을 확인하는 방식을 정의할 수 있습니다. SAML 1.1 의 경우 어설션만 확인하도록 선택할 수 있습니다.

SAML 1.1 과 WS-페더레이션은 암호화를 지원하지 않습니다.

인증서 데이터 저장소에 여러 개의 개인 키와 인증서가 있을 수 있습니다. 여러 개의 페더레이션된 파트너가 있는 경우 각 파트너마다 다른 키 쌍을 사용할 수 있습니다.

참고: 시스템이 FIPS_COMPAT 또는 FIPS_MIGRATE 모드에서 작동하는 경우 풀다운 목록에서 모든 인증서와 키 항목을 사용할 수 있습니다. 시스템이 FIPS 전용 모드에서 작동하는 경우에는 FIPS 승인 인증서 및 키 항목만 사용할 수 있습니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. 수정하려는 신뢰 당사자-어설션 당사자 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "서명" 단계로 이동합니다.
4. "확인 인증서 별칭" 필드에서 인증서 데이터 저장소의 별칭을 선택합니다.

   이 필드에 데이터를 입력하면 서명된 어설션, 응답 또는 둘 다를 확인하는 인증서가 지정됩니다. 인증서 데이터 저장소에 인증서가 없는 경우 "가져오기"를 클릭하여 가져옵니다. 또는 "생성"을 클릭하여 인증서 요청을 생성합니다.

참고: 테스트 환경에서 제품을 사용 중인 경우 테스트를 단순화하기 위해 서명 처리를 중지할 수 있습니다. "서명 처리 사용 안 함" 확인란을 클릭하십시오.

서명 구성이 완료되었습니다.

SAML 2.0 IdP 에서의 서명 구성

파트너 관계 마법사의 "서명 및 암호화" 단계에서는 제품에서 다음의 서명 기능에 개인 키 및 인증서가 사용되는 방식을 정의할 수 있습니다.

- SAML 어설션, 어설션 응답 및 인증 요청을 서명하고 확인합니다.

  SAML 2.0 POST 바인딩의 경우 어설션에 서명해야 합니다.
- 싱글 로그아웃 응답 및 요청에 서명합니다(HTTP-리디렉션 및 SOAP 바인딩).

인증서 데이터 저장소에 여러 개의 개인 키와 인증서가 있을 수 있습니다. 여러 개의 페더레이션된 파트너가 있는 경우 각 파트너마다 다른 키 쌍을 사용할 수 있습니다.

참고: 시스템이 FIPS_COMPAT 또는 FIPS_MIGRATE 모드에서 작동하는 경우 풀다운 목록에서 모든 인증서와 키 항목을 사용할 수 있습니다. 시스템이 FIPS 전용 모드에서 작동하는 경우에는 FIPS 승인 인증서 및 키 항목만 사용할 수 있습니다.

서명 옵션을 구성하려면

1. 파트너 관계 마법사에서 "서명 및 암호화" 단계를 선택합니다.
2. "서명" 섹션에서 "서명 개인 키 별칭" 필드에 대한 별칭을 선택합니다. 사용 가능한 개인 키가 없는 경우 "가져오기"를 클릭하여 가져오십시오. 또는 "생성"을 클릭하여 인증서 요청을 생성하십시오.

   이 필드에 데이터를 입력하면 어설션 당사자가 어설션, 싱글 로그아웃 요청 및 응답에 서명하기 위해 사용하는 개인 키가 지정됩니다.

   참고: 필드의 설명을 보려면 "도움말"을 클릭하십시오.
3. "서명 알고리즘" 필드에서 디지털 서명에 대한 해시 알고리즘을 선택합니다. IdP 는 지정된 알고리즘을 사용하여 어설션, 응답 및 SLO-SOAP 메시지에 서명합니다.

   응용 프로그램에 가장 적합한 알고리즘을 선택하십시오.

   RSAwithSHA256 이 RSAwithSHA1 보다 결과 암호화 해시 값에 사용되는 비트 수가 많으므로 더 안전합니다.

   선택하는 알고리즘은 시스템의 모든 서명 기능에 사용됩니다.
4. 인증서 데이터 저장소에서 또는 "확인 인증서 별칭" 필드에서 별칭을 선택합니다.

   이 필드에 데이터를 입력하면 서명된 인증 요청이나 싱글 로그아웃 요청 또는 응답을 확인하는 인증서가 지정됩니다. 데이터베이스에 인증서가 없는 경우 "가져오기"를 클릭하여 가져오십시오.
5. (선택 사항) 어설션이나 응답 또는 둘 다에 대한 아티팩트 및 POST 서명 옵션을 지정합니다.
6. (선택 사항) 싱글 로그아웃을 사용할 때 로그아웃 요청, 로그아웃 응답 또는 둘 다에 대한 SLO SOAP 서명 옵션을 지정합니다.
7. (선택 사항) "서명된 인증 요청 필요"에 대한 확인란을 선택합니다. 이 확인란은 어설션 당사자가 신뢰 당사자의 서명된 요청만 수락하는 것을 확인합니다.

   파트너 관계를 활성화하여 구성 변경 내용을 모두 적용하고 파트너 관계를 사용 가능한 상태로 만듭니다. 서비스를 다시 시작하는 것만으로는 충분하지 않습니다.

테스트 환경에서 제품을 사용 중인 경우 테스트를 단순화하기 위해 서명 처리를 중지할 수 있습니다. "서명 처리 사용 안 함" 확인란을 클릭하십시오.

중요! SAML 2.0 프로덕션 환경에서는 서명 처리가 사용되도록 설정하십시오.

SAML 2.0 IdP 에서의 암호화 구성

파트너 관계 마법사의 "서명 및 암호화" 단계에서는 정책 서버가 다음 태스크를 수행하기 위해 개인 키 및 인증서를 사용하는 방법을 정의할 수 있습니다.

- SAML 어설션, 어설션 응답 및 인증 요청을 서명하고 확인합니다.

  SAML 2.0 POST 바인딩의 경우 어설션에 서명해야 합니다.
- 싱글 로그아웃 응답 및 요청에 서명합니다(HTTP-리디렉션 및 SOAP 바인딩).
- 전체 어설션, 이름 ID 및 특성을 암호화 및 암호 해독합니다.

인증서 데이터 저장소에 여러 개의 개인 키와 인증서가 있을 수 있습니다. 여러 개의 페더레이션된 파트너가 있는 경우 각 파트너마다 다른 키 쌍을 사용할 수 있습니다.

암호화 옵션을 구성하려면

1. "암호화" 섹션에서 다음 확인란 중 하나 또는 둘 다를 선택하여 암호화할 어설션 데이터를 지정합니다.
   - 이름 ID 암호화
   - 어설션 암호화
2. "암호화 인증서 별칭"에서 인증서 데이터 저장소의 인증서 별칭을 선택합니다.

   이 인증서는 어설션 데이터를 암호화합니다. 사용할 수 있는 인증서가 없는 경우에는 "가져오기"를 클릭하여 가져옵니다.

3. "암호화 블록 알고리즘" 및 "암호화 키 알고리즘" 필드에 대한 값을 선택합니다.

   다음의 블록/키 알고리즘 조합에서 인증에 필요한 최소 키 크기는 1024 비트입니다.

   - 암호화 블록 알고리즘: 3DES

     암호화 키 알고리즘: RSA-OEAP

   - 암호화 블록 알고리즘: AES-256

     암호화 키 알고리즘: RSA-OEAP

     참고: AES-256 비트 암호화 블록 알고리즘을 사용하려면 JCE(Java Cryptography Extension) Unlimited Strength Jurisdiction Policy Files 를 설치하십시오. 이 파일은 http://java.sun.com/javase/downloads/index.jsp 에서 다운로드할 수 있습니다.

암호화 구성이 완료되었습니다.

SAML 2.0 SP 에서의 서명 구성

파트너 관계 마법사의 "서명 및 암호화" 단계에서는 정책 서버가 다음 태스크를 수행하기 위해 개인 키 및 인증서를 사용하는 방법을 정의할 수 있습니다.

- SAML 어설션 서명 및 어설션 응답을 확인하고 인증 요청에 서명합니다.

  참고: SAML 2.0 POST 바인딩의 경우 IdP 가 어설션에 서명해야 합니다.

- 싱글 로그아웃 응답 및 요청에 서명합니다(HTTP-리디렉션 및 SOAP 바인딩).

인증서 데이터 저장소에 여러 개의 개인 키와 인증서가 있을 수 있습니다. 여러 개의 페더레이션된 파트너가 있는 경우 각 파트너마다 다른 키 쌍을 사용할 수 있습니다.

참고: 시스템이 FIPS_COMPAT 또는 FIPS_MIGRATE 모드에서 작동하는 경우 풀다운 목록에서 모든 인증서와 키 항목을 사용할 수 있습니다. 시스템이 FIPS 전용 모드에서 작동하는 경우에는 FIPS 승인 인증서 및 키 항목만 사용할 수 있습니다.

서명 옵션을 구성하려면

1. 파트너 관계 마법사에서 "서명 및 암호화" 단계를 선택하여 시작합니다.
2. "서명" 섹션의 "서명 개인 키 별칭" 필드에서 인증서 데이터 저장소의 별칭을 선택합니다. 데이터베이스에 개인 키가 없는 경우 "가져오기"를 클릭하여 가져옵니다. 또는 "생성"을 클릭하여 키 쌍을 만들고 인증서 요청을 생성합니다.

   이 필드에 데이터를 입력하면 신뢰 당사자가 인증 요청과 싱글 로그아웃 요청 및 응답에 서명하는 데 사용하는 개인 키가 지정됩니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
3. "서명 알고리즘" 필드에서 디지털 서명에 대한 해시 알고리즘을 선택합니다. SP 는 지정된 알고리즘을 사용하여 인증 요청 및 SLO-SOAP 메시지에 서명합니다.

   응용 프로그램에 가장 적합한 알고리즘을 선택하십시오.

   RSAwithSHA256 이 RSAwithSHA1 보다 결과 암호화 해시 값에 사용되는 비트 수가 많으므로 더 안전합니다.

   SiteMinder 는 선택하는 알고리즘을 모든 서명 기능에 사용합니다.
4. "확인 인증서 별칭" 필드에서 인증서 데이터 저장소의 별칭을 선택합니다.

   이 필드에 데이터를 입력하면 신뢰 당사자가 서명된 어설션이나 싱글 로그아웃 요청 및 응답을 확인하는 데 사용하는 인증서가 지정됩니다. 데이터베이스에 인증서가 없는 경우 "가져오기"를 클릭하여 가져오십시오.
5. (선택 사항) SP 가 모든 인증 요청에 서명하도록 하려면 "서명 인증 요청"을 선택합니다. 원격 어설션 당사자가 인증 요청에 서명을 요구하는 경우 이 옵션을 선택합니다.

   파트너 관계를 활성화하여 구성 변경 내용을 모두 적용하고 파트너 관계를 사용 가능한 상태로 만듭니다. 서비스를 다시 시작하는 것만으로는 충분하지 않습니다.

테스트 환경에서 SiteMinder 를 사용 중인 경우 테스트를 단순화하기 위해 서명 처리를 중지할 수 있습니다. 기능을 사용하지 않으려면 "서명 처리 사용 안 함" 확인란을 클릭하십시오.

중요! SAML 2.0 프로덕션 환경에서는 서명 처리가 사용되도록 설정하십시오.

SAML 2.0 SP 에서의 암호화 구성

"서명 및 암호화" 단계에서는 어설션, 이름 ID, 특성의 암호화 및 암호 해독을 포함하여 SP 가 개인 키 및 인증서를 사용하는 방법을 구성할 수 있습니다.

인증서 데이터 저장소에 여러 개의 개인 키와 인증서가 있을 수 있습니다. 여러 개의 페더레이션된 파트너가 있는 경우 각 파트너마다 다른 키 쌍을 사용할 수 있습니다.

참고: 시스템이 FIPS_COMPAT 또는 FIPS_MIGRATE 모드에서 작동하는 경우 풀다운 목록에서 모든 인증서와 키 항목을 사용할 수 있습니다. 시스템이 FIPS 전용 모드에서 작동하는 경우에는 FIPS 승인 인증서 및 키 항목만 사용할 수 있습니다.

암호화 옵션을 구성하려면

1. "암호화" 섹션에서 다음 확인란 중 하나 또는 둘 다를 선택하여 어설션에서 올바른 데이터가 암호화되도록 합니다.
   - 암호화된 이름 ID 필요
   - 암호화된 어설션 필요

   참고: AES-256 비트 암호화 블록 알고리즘을 사용하려면 Sun JCE(Java Cryptography Extension) Unlimited Strength Jurisdiction Policy Files 를 설치하십시오. 이 파일은 http://java.sun.com/javase/downloads/index.jsp 에서 다운로드할 수 있습니다.
2. "암호 해독 개인 키 별칭"에 대해 인증서 데이터 저장소의 별칭을 선택합니다.

   이 개인 키는 암호화된 어설션 데이터를 암호 해독합니다. 사용할 수 있는 인증서가 없는 경우 "가져오기"를 클릭하여 가져오거나 "생성"을 클릭하여 키 쌍을 만들고 인증서 요청을 생성합니다.

암호화 구성이 완료되었습니다.

제 19 장: 페더레이션 환경 보호

이 섹션은 다음 항목을 포함하고 있습니다.

페더레이션된 트랜잭션의 보안 방법

어설션 암호화 및 파트너 사이트 간에 SSL 연결 사용 등과 같이 페더레이션된 파트너 간의 트랜잭션에 보안을 적용하는 데 도움이 되는 몇 가지 메커니즘이 있습니다.

파트너 관계 페더레이션을 사용하여 페더레이션 환경을 설정하는 경우 환경을 보호하기 위한 몇 가지 권장 사항은 다음과 같습니다.

- 한 번만 사용할 어설션을 생성합니다.
- 교차 사이트 스크립팅을 방지합니다.

이러한 항목은 뒤에 나오는 단원에서 설명합니다.

어설션의 일회 사용 적용

유효 기간이 지난 어설션을 재사용하면 오래된 아이덴티티 정보를 사용하여 인증 결정이 내려집니다. 이러한 재사용을 방지하기 위해 SiteMinder 는 SAML 1.x 및 2.0 사양에 따라 일회용 어설션을 생성할 수 있습니다. 어설션 재사용으로 인한 문제를 방지하기 위해, 어설션에는 신뢰 당사자에게 이후 트랜잭션을 위해 어설션을 보존하지 않도록 지정하는 요소가 포함됩니다.

SiteMinder 가 어설션 당사자(생산자/IdP)로 작동하는 경우 어설션의 일회 사용을 구성할 수 있습니다. SAML 1.x 생산자의 경우 **DoNotCache 조건 설정** 설정을 선택할 수 있습니다. SAML 2.0 IdP 의 경우 **OneTimeUse 조건 설정** 설정을 선택할 수 있습니다. 이러한 구성 설정을 둘 다 사용하면 SiteMinder 가 일회 사용 조건을 나타내는 적절한 요소를 어설션에 삽입할 수 있습니다.

참고: 어설션의 일회 사용과 SAML 1.x 및 2.0 HTTP-POST 싱글 사인온에 대한 단일 사용 정책을 혼동하지 마십시오. SiteMinder 는 신뢰 당사자로 작동할 때 단일 사용 정책을 사용하며 이는 POST 트랜잭션 전용입니다. 일회 사용 기능은 HTTP-아티팩트 및 HTTP-POST 용입니다.

페더레이션 환경의 연결 보안

페더레이션된 파트너 간에 전송되거나 파트너와 응용 프로그램 간에 전송되는 아이덴티티 정보는 통신이 보안 연결을 통해 수행될 때 최적으로 보호됩니다.

신뢰 당사자와 대상 응용 프로그램 간의 연결 보안

신뢰 당사자 측에서 클라이언트 사이트의 대상 응용 프로그램으로 흐르는 데이터 전송을 보호하십시오. 보안 연결을 통신 채널로 사용하면 보안 공격에 대한 환경 취약점이 줄어듭니다.

예를 들어 신뢰 당사자가 추출하여 클라이언트 응용 프로그램에 보내는 특성이 어설션에 포함될 수 있습니다. 신뢰 당사자는 HTTP 헤더 변수나 쿠키를 사용하여 이러한 특성을 응용 프로그램에 전달할 수 있습니다. 헤더나 쿠키에 저장된 특성이 클라이언트 측에서 덮어쓰여질 수 있으므로 악의적인 사용자가 다른 사용자를 가장할 수 있습니다. SSL 연결을 사용하면 환경이 이러한 종류의 보안 위반으로부터 보호됩니다.

해당 ACO(에이전트 구성 개체)에서 UseSecureCookies 매개 변수를 설정하여 이 취약점을 방지하는 것이 가장 좋습니다. UseSecureCookies 매개 변수는 페더레이션 웹 서비스에 "secure" 플래그로 표시된 쿠키를 생성하도록 지시합니다. 이 플래그는 쿠키가 SSL 통신 채널을 통해서만 전송됨을 나타냅니다.

참고: 수정할 ACO 는 페더레이션 환경 설정에 따라 다릅니다. 웹 에이전트가 설치된 것과 같은 시스템에 페더레이션 웹 서비스를 배포하는 경우 웹 에이전트에 대한 ACO 를 편집하십시오. 웹 에이전트가 설치된 것과 다른 시스템에 페더레이션 웹 서비스를 배포하는 경우에는 페더레이션 웹 서비스에 대해 생성한 고유한 ACO 를 편집하십시오.

SiteMinder 어설션 당사자의 초기 인증 보안

SiteMinder 어설션 당사자의 초기 사용자 인증에는 잠재적인 취약점이 있습니다. 사용자가 어설션 당사자 측에서 사용자 세션을 설정하기 위해 맨 처음 인증할 때 세션 ID 쿠키가 브라우저에 기록됩니다. 쿠키가 비 SSL 연결을 통해 전송되면 공격자가 쿠키를 획득하고 중요한 사용자 정보를 도용할 수 있습니다. 그런 다음 공격자가 이 정보를 사용하여 사용자를 가장하거나 아이덴티티를 도용할 수 있습니다.

에이전트 구성 개체에서 수정할 수 있는 웹 에이전트 매개 변수 UseSecureCookies 를 설정하여 이 취약점을 방지하는 것이 가장 좋습니다. UseSecureCookies 매개 변수는 웹 에이전트에 "secure" 플래그로 표시된 쿠키를 생성하도록 지시합니다. 이 플래그는 브라우저가 SSL 연결을 통해서만 쿠키를 전달하여 보안이 강화됨을 나타냅니다. 일반적으로 모든 URL 에 대해 SSL 연결을 설정하는 것이 좋습니다.

페더레이션된 네트워크를 교차 사이트 스트립팅으로부터 보호

응용 프로그램이 브라우저의 입력 텍스트를 표시할 경우 XSS(교차 사이트 스크립팅) 공격이 발생할 수 있습니다. 응용 프로그램은 실행 가능한 스크립트를 구성할 수 있는 문자를 테스트하지 못할 수 있습니다. 이러한 문자가 표시되면 원치 않는 스크립트가 브라우저에서 실행될 수 있습니다.

SiteMinder 는 페더레이션 기능에 사용할 여러 JSP 를 제공합니다. 이러한 JSP 는 요청에 있는 문자를 확인하여 출력 스트림의 안전하지 않은 정보가 브라우저에 표시되지 않도록 합니다.

SiteMinder 가 요청을 받으면 다음 JSP 가 디코딩된 값을 검사하여 교차 사이트 스크립팅 문자가 있는지 확인합니다.

- idpdiscovery.jsp

  신뢰 당사자 측에서 아이덴티티 공급자 검색에 사용됩니다.

- linkaccount.jsp

  신뢰 당사자 측에서 동적 계정 연결에 사용됩니다.

- sample_application.jsp

  IDP 에서 싱글 사인온을 시작하는 데 사용됩니다. 이 샘플 응용 프로그램을 사용하여 사용자를 먼저 SSO 서비스에 연결한 다음 사용자 지정 웹 응용 프로그램에 연결할 수 있습니다. 일반적으로 사용자 고유의 응용 프로그램을 사용합니다.

- signoutconfirmurl.jsp

  계정 파트너에서 WS-페더레이션 사인아웃에 사용됩니다.

- unsolicited_application.jsp

  사용자가 SSO 서비스에 먼저 전송되지 않고 웹 응용 프로그램에 직접 전송될 때 IdP 에서 시작되는 싱글 사인온에 사용됩니다.

해당 페이지에서는 요청을 검사하여 다음 문자가 있는지 확인합니다.

문자	설명
<	왼쪽 꺾쇠 괄호
>	오른쪽 꺾쇠 괄호
‘	작은따옴표
"	큰따옴표
%	백분율 기호
;	세미콜론
(	여는(왼쪽) 괄호
)	닫는(오른쪽) 괄호
&	앰퍼샌드
+	더하기 기호

각 JSP 에는 검사할 문자를 정의하는 변수가 포함되어 있습니다. 문자 집합을 확장하려면 이러한 JSP 를 수정하십시오.

제 20 장: 신뢰 당사자에서의 응용 프로그램 통합

이 섹션은 다음 항목을 포함하고 있습니다.

신뢰 당사자와 응용 프로그램의 상호 작용

파트너 관계 마법사의 "응용 프로그램 통합" 단계는 신뢰 당사자 측에서만 적용됩니다. 이 단계에서는 사용자 신원을 확인하고 사용자를 대상 응용 프로그램으로 연결하기 위한 페더레이션된 작업의 다양한 측면을 정의할 수 있습니다.

"응용 프로그램 통합" 단계에서 구성할 수 있는 기능은 다음과 같습니다.

- 사용자를 대상 응용 프로그램으로 리디렉션
- 어설션 특성을 응용 프로그램 특성에 매핑(SAML 만 해당)
- 사용자 아이덴티티 프로비저닝
- 인증 실패 시 사용자 리디렉션

사용자를 대상 응용 프로그램으로 리디렉션

"응용 프로그램 통합" 단계의 "대상 응용 프로그램" 섹션에서는 사용자를 대상 응용 프로그램으로 리디렉션하는 방법을 정의할 수 있습니다. 선택하는 리디렉션 방법은 사용자와 함께 대상 응용 프로그램에 전달하고자 하는 데이터의 유형에 따라 달라집니다.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 "응용 프로그램 통합" 단계로 이동합니다.
2. "리디렉션 모드" 필드에서 리디렉션 방법을 선택합니다. 다음 정보에 주의하십시오.
   - "쿠키 데이터"를 선택하면 "URL 인코드 특성 쿠키 데이터" 확인란을 선택하여 쿠키의 특성 데이터를 URL 인코딩할 수 있습니다. 이 옵션은 SAML 1.1 및 2.0 에만 사용할 수 있습니다.
   - "개방 형식 쿠키" 또는 "개방 형식 쿠키 게시" 옵션을 선택하는 경우 추가 필수 설정 및 옵션 설정을 구성하십시오. 개방 형식 쿠키와는 달리, 개방 형식 쿠키 게시는 HTTP-POST 요청 형식으로 데이터를 보냅니다.

     신뢰 당사자가 여러 특성 값이 있는 어설션을 받는 경우 정책 서버는 모든 값을 대상 응용 프로그램에 쿠키로 전달합니다.
   - FIPS 호환 알고리즘 중 하나(AES 알고리즘)를 선택하는 경우 CA SiteMinder® Federation SDK 를 사용하여 개방 형식 쿠키를 생성하십시오. .NET SDK 를 사용하는 경우에는 AES128/CBC/PKCS5Padding 암호화 알고리즘만 사용하십시오.

     대상 응용 프로그램은 쿠키를 만드는 SDK 와 동일한 언어를 사용해야 합니다. CA SiteMinder® Federation Java SDK 를 사용하는 경우 응용 프로그램은 Java 로 작성되어야 합니다. .NET SDK 를 사용하는 경우 응용 프로그램이 .NET 을 지원해야 합니다.
   - 리디렉션 모드로 "HTTP 헤더"를 선택하는 경우 SiteMinder 는 하나의 헤더에 여러 특성 값을 제공할 수 있습니다. 각 특성 값을 쉼표로 구분하십시오. 이 옵션은 SAML 1.1 및 2.0 에만 사용할 수 있습니다.

     HTTP 헤더를 리디렉션 모드로 사용 (페이지 275)하는 방법과 헤더를 보호하는 방법에 대한 자세한 내용을 참조하십시오.

   필드의 설명을 보려면 "도움말"을 클릭하십시오.

3. "대상" 필드에 대상 응용 프로그램의 URL 을 입력합니다.

   대상 리소스가 있는 서버의 앞에 프록시가 있는 경우 프록시 호스트의 URL 을 입력합니다. 프록시는 모든 페더레이션 요청을 로컬로 처리합니다. 프록시 호스트는 대상 서버의 앞에 있는 모든 시스템이 될 수 있습니다. SiteMinder 가 인터넷에서 직접 액세스되는 경우에는 SiteMinder 자체가 프록시 호스트가 될 수도 있습니다. 프록시를 사용할 때는 대상으로 지정하는 URL 이 SiteMinder 를 통과해야 합니다. 예를 들어 기준 URL 이 fed.demo.com 이고 백엔드 서버 리소스가 mytarget/target.jsp 인 경우 이 필드의 값은 http://fed.demo.com:5555/mytarget/target.jsp 입니다.

   SAML 2.0 에서는 RelayState 쿼리 매개 변수로 이 필드를 재정의하는 경우 이 필드를 비워 둘 수 있습니다. RelayState 쿼리 매개 변수는 싱글 사인온을 트리거하는 URL 의 일부일 수 있습니다. 이렇게 설정하려면 "릴레이 상태가 대상 무시" 확인란을 선택하십시오.

대상으로의 리디렉션이 설정되었습니다.

HTTP 헤더를 사용하여 어설션 데이터 전달(SAML 만 해당)

SAML 엔터티의 경우 정책 서버는 HTTP 헤더를 사용하여 어설션에서 백엔드 응용 프로그램으로 아이덴티티 특성을 전달할 수 있습니다. 백엔드 응용 프로그램은 싱글 사인온의 대상 응용 프로그램 또는 사용자 프로비저닝 응용 프로그램일 수 있습니다. 이 헤더는 암호화된 쿠키에 포함되어 전달됩니다.

헤더의 이름은 어설션 특성과 동일합니다. 예를 들어 어설션 특성이 "address"인 경우 응용 프로그램은 "ADDRESS"라는 HTTP 헤더를 찾습니다.

어설션 특성은 대/소문자를 구분하지만 HTTP 헤더는 그렇지 않습니다. 정책 서버는 대/소문자만 다른 동일한 특성을 전달한 다음 이를 HTTP 헤더에 매핑할 수 없습니다. 예를 들어 "address"와 "Address"가 동시에 헤더로 전달될 수 없습니다. 일반적으로 대/소문자나 형식만 다르고 이름이 동일한 특성을 사용하지 마십시오.

다음의 추가적인 값이 헤더로 전달됩니다.

- NAMEID
- FORMAT
- AUTHNCONTEXT

HTTP 헤더 보호

인증되지 않은 사용자가 어설션 특성의 이름을 알고 있는 경우 이 사용자는 이 이름을 브라우저에서 헤더로 설정할 수 있습니다. 헤더가 설정되면 악의적인 사용자가 대상 응용 프로그램에 대한 액세스 권한을 획득할 수 있습니다. 대상 응용 프로그램은 예기치 않은 헤더 값을 발견하고 SiteMinder 의 어설션 소비 없이 리소스에 대한 액세스를 허용합니다.

FedHeaderPrefix 에 대한 값을 설정하면 다음과 같은 시나리오가 방지됩니다.

1. 인증되지 않은 사용자가 HTTP 헤더의 이름을 알아냅니다. 이 헤더 이름에는 접두사가 포함되어 있습니다.
2. 악의적인 사용자가 헤더를 포함하여 들어오는 요청을 정책 서버에 전송합니다.
3. 정책 서버는 접두사를 포함한 헤더의 출처가 들어오는 요청이고 이 헤더가 내부에서 생성되지 않은 것으로 인식하므로 이를 제거합니다.
4. 시스템에서는 내부에서 생성한 정상 헤더를 백엔드 응용 프로그램에 전달하기 전에 각 헤더에 지정된 접두사를 추가합니다. 그런 다음 이 헤더가 응용 프로그램에 전달됩니다.

어설션 데이터를 전달하도록 HTTP 헤더 구성(SAML 만 해당)

SiteMinder 는 HTTP 헤더를 사용하여 어설션 데이터를 전달할 수 있습니다.

다음 단계를 수행하십시오.

1. 페더레이션 트래픽을 처리하는 신뢰 당사자 시스템에 SiteMinder 웹 에이전트가 설치되어 있는지 확인합니다.
2. *web_agent_home*/conf 로 이동하여 WebAgent.conf 파일을 수정합니다. 다음과 같이 다음 항목의 주석 처리를 제거합니다.

   Windows

   LoadPlugin="*path*\SAMLDataPlugin.dll"

   UNIX

   LoadPlugin="*path*/SAMLDataPlugin.so"

3. (선택 사항이지만 권장하지 않음) 적절한 에이전트 구성 개체 또는 웹 에이전트에 **fedheaderprefix** 설정을 추가합니다. 임의의 문자열을 접두사로 입력합니다.

   fedheaderprefix 설정은 SiteMinder 가 HTTP 헤더에 추가하는 전역 접두사를 지정합니다. 접두사를 설정하면 SiteMinder 가 어설션을 소비하기 전에 인증되지 않은 사용자가 HTTP 헤더를 조작하지 못하게 됩니다. 따라서 정상 헤더만 대상 응용 프로그램으로 전달됩니다. HTTP 헤더 보호 (페이지 276)에 대한 자세한 내용을 참조하십시오.

4. 파트너 관계 마법사의 "응용 프로그램 통합" 단계에서 다음 태스크 중 *하나*를 수행합니다.

   - "HTTP 헤더"를 대상 응용 프로그램에 대한 "리디렉션 모드"로 선택합니다.
   - "HTTP 헤더"를 사용자 프로비저닝에 대한 "전송 옵션"으로 선택합니다.

이제 HTTP 헤더가 특성 데이터를 전달하도록 구성되었습니다.

어설션 특성을 응용 프로그램 특성에 매핑(SAML 만 해당)

SAML 1.1 소비자 또는 SAML 2.0 SP 에서 어설션 특성 집합을 나가는 응용 프로그램 특성 집합에 매핑할 수 있습니다. 그런 다음 응용 프로그램 특성이 대상 응용 프로그램으로 전송됩니다. 특성 매핑을 사용하면 대상 응용 프로그램을 수정할 필요 없이 사용자에게 사용자 지정된 환경을 제공할 수 있습니다. 특성은 파트너 관계 단위로 매핑되므로 신뢰 당사자 측 응용 프로그램을 여러 어설션 당사자에 사용할 수 있습니다.

다음과 같은 유형의 매핑을 사용할 수 있습니다.

- 어설션 특성 이름을 응용 프로그램 특성 이름으로 변환합니다.

  예

  들어오는 어설션 특성이 Region=US 일 수 있습니다. 특성은 나가는 응용 프로그램 특성 ServiceLocation=US 로 변환될 수 있습니다.

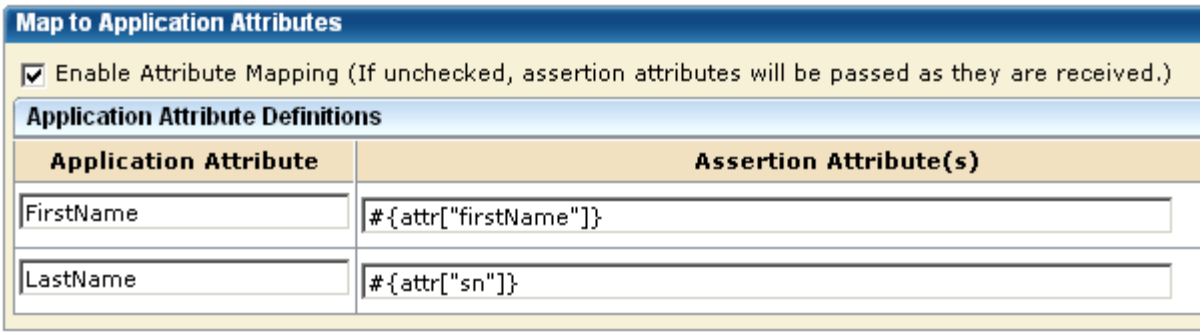
- 개별적인 특성과 해당 값을 단일 특성으로 전환합니다.

  예

  어설션에 Name=Bob 및 LastName=Smith 의 두 특성이 포함되어 있습니다. 이 두 특성을 FullName =Bob Smith 로 변환할 수 있습니다.

응용 프로그램 특성 정의 테이블 사용

"응용 프로그램 통합" 대화 상자의 "응용 프로그램 특성 정의" 테이블에서 특성 매핑 규칙을 정의할 수 있습니다. 이 테이블은 다음 그림에 표시되어 있습니다.



"응용 프로그램 특성" 및 "어설션 특성" 열은 원격 생산자 또는 IdP 엔터티에 대한 어설션 특성을 사용하여 채워집니다. 이러한 특성은 이 로컬 신뢰 당사자 측에서 구성하십시오. "어설션 특성" 열에 대해서는 어설션 특성 이름이 입력됩니다. 이에 해당하는 UEL(Unified Expression Language) 문자열이 "어설션 특성" 열에 입력됩니다.

신뢰 당사자의 관리자 또는 응용 프로그램 통합자는 특성 매핑을 구성하기 위해 다음 정보를 알아야 합니다.

- 대상 응용 프로그램 특성의 이름
- 어설션의 특성 이름
- 어설션 특성과 대상 응용 프로그램 특성 간의 매핑 관계. 매핑 관계를 안다는 것은 사용 가능한 어설션 특성을 필요한 응용 프로그램 특성으로 전환하는 방법을 안다는 것을 의미합니다.

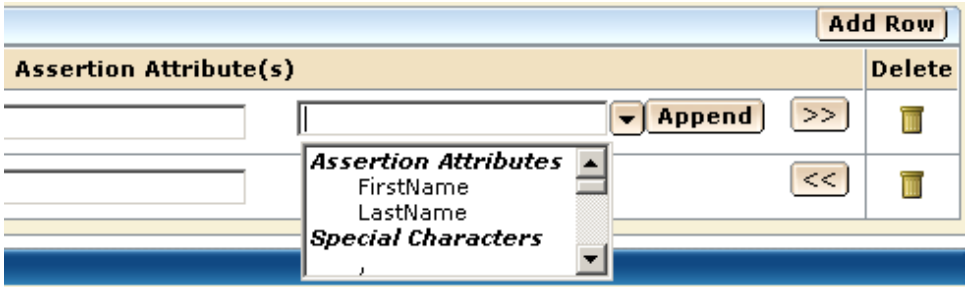
특성 매핑을 설정하기 전에 필요한 당사자 측에서 응용 프로그램 및 어설션 특성의 이름을 수집하십시오.

응용 프로그램 특성은 대상 응용 프로그램이 사용하는 특성을 반영해야 하므로 응용 프로그램에 맞게 기본값을 수정해야 합니다. 응용 프로그램 관리자와의 대역 외 통신을 통해 응용 프로그램 특성을 가져올 수 있습니다.

식 작성기를 사용하여 매핑 규칙 작성

UI 에서는 매핑 규칙을 작성하는 데 유용한 식 작성기를 제공합니다. 식 작성기에 액세스하려면 "어설션 특성" 필드 오른쪽의 슬라이더 단추(<<)를 선택하십시오. 슬라이더 단추를 선택하면 빈 필드와 풀다운 화살표가 표시됩니다. 화살표를 선택하면 매핑 구성에 사용할 수 있는 어설션 특성과 특수 문자의 목록이 표시됩니다. 식 작성기를 숨기려면 슬라이더 단추(>>)를 클릭하십시오.

다음 그림에서는 식 작성기 메뉴를 보여 줍니다.



식 작성기의 "어설션 특성" 목록은 원격 생산자 또는 IdP 엔터티에 대한 어설션 특성으로 채워집니다. 이러한 특성은 이 로컬 신뢰 당사자 측에서 구성하십시오. 특성이 어설션에 있다는 점을 알고 있으면 항목을 수동으로 지정할 수 있습니다. 식 작성기 메뉴의 옵션만 사용할 필요는 없습니다.

"특수 문자" 목록에는 매핑 규칙을 작성하는 데 사용할 수 있는 쉼표와 백분율 기호 등의 문자가 포함되어 있습니다. 목록에서 문자를 선택하거나 문자를 수동으로 입력할 수 있습니다.

중요! 이 테이블에 어설션 특성을 입력할 때 어설션 특성은 원격 어설션 당사자 측에서 지정된 어설션 특성을 기준으로 대/소문자가 구분됩니다. 대/소문자가 일치해야 합니다. SiteMinder 가 파트너 관계의 양쪽 모두에 있는 경우 특성은 원격 IdP 파트너 관계 마법사의 이름 ID 및 특성 단계에서 지정됩니다. 파트너와의 대역 외 통신에서 또는 메타데이터 가져오기를 통해 어설션 특성을 가져오십시오.

매핑 규칙이 정의되면 SiteMinder 는 데이터를 레거시 쿠키, 개방 형식 쿠키 또는 HTTP 헤더에 넣습니다. 그런 다음 SiteMinder 는 데이터를 응용 프로그램으로 전송합니다. "응용 프로그램 통합" 대화 상자의 "대상 응용 프로그램" 섹션에서 전송 방법을 지정하십시오.

매핑 수정 및 삭제

언제든지 "응용 프로그램 특성 정의" 테이블에서 특성 매핑을 변경하거나 제거할 수 있습니다.

매핑을 수정하려면

1. 커서를 수정할 행의 필드에 놓은 다음 새 텍스트를 입력합니다. 식 작성기를 사용하여 현재 식의 끝에 값을 더 추가할 수도 있습니다.
2. "다음"을 클릭하고 마법사의 마지막 단계로 진행하여 변경 내용을 저장합니다.

매핑을 삭제하려면

1. 제거할 항목의 "삭제" 열에서 휴지통을 클릭합니다.
2. "다음"을 클릭하고 마법사의 마지막 단계로 진행하여 변경 내용을 저장합니다.

적절한 구문을 사용하여 특성 매핑 규칙 작성

특성 매핑은 어설션 특성을 응용 프로그램 특성으로 전환하는 매핑 규칙을 사용합니다. 특성 매핑이 사용되도록 지정하면 SiteMinder 에서 기본 매핑 규칙을 생성합니다. 규칙은 원격 생산자 또는 IdP 엔터티에 대해 지정된 어설션 특성을 기반으로 합니다. 이러한 구성 태스크는 모두 로컬 신뢰 당사자 측에서 수행됩니다. 특성 매핑이 사용되지 않도록 설정하면 어설션 특성은 대상 응용 프로그램에 "있는 그대로" 전달됩니다.

SiteMinder 는 JSP 및 JSF 와 비슷한 UEL(Unified Expression Language) 구문을 매핑에 사용합니다. 각 어설션 특성이 hashmap 에 들어가고 **attr** 키워드가 할당됩니다. UEL 식 계산기가 매핑 규칙 목록을 순환하여 어설션 특성의 hashmap 에 이를 적용합니다. 그런 후 식 계산기는 결과 응용 프로그램 특성이 포함된 다른 hashmap 을 생성합니다. 나가는 응용 프로그램 특성의 hashmap 은 쿠키 콘텐츠 또는 헤더 변수로 변환되어 대상 응용 프로그램으로 전송됩니다.

식을 작성하려면 SiteMinder 가 식에 사용하는 구문을 이해하는 것이 중요합니다.

단일 특성 표현

단일 어설션 특성을 표현하려면 다음 구문을 사용하십시오.

#{attr["*attribute_name*"]}

예: #{attr["Name"]}는 "이름" 어설션 특성의 값을 나타냅니다.

복합 특성 표현

값 식을 연결하여 복합 값을 구성할 수 있습니다(선택적 구분 기호 사용). 복합 어설션 특성을 표현하려면 다음 구문을 사용하십시오.

#{attr["*first_attribute*"]}*optional_character* #{attr["*second_attribute*"]}

매핑 예

다음은 매핑 규칙의 예입니다. 이러한 예는 다음 형식으로 표시됩니다.

application_attribute=assertion_attributes_expression

이름 예

구문

ID = #{attr["Name"]}

샘플 결과

BobSmith

단순 연결 예

구문

FullName = #{attr["FirstName"]},#{attr["LastName"]}

샘플 결과

Bob,Smith

구문

FullName = #{attr["LastName"]},#{attr["FirstName"]}

샘플 결과

Smith,Bob

공백은 특수 문자로 간주됩니다. 식에서 특성 사이에 공백을 넣으려면 공백을 입력하십시오. 예를 들면 다음과 같습니다.

구문

FullName = #{attr["LastName"]}, #{attr["FirstName"]}

샘플 결과

Smith, Bob

날짜 예

구문

Date = #{attr["month"]}/#{attr["dateOfMonth"]}/#{attr["year"]}

샘플 결과

01/05/2010

구문

Date = #{attr["monthSymbol"]} #{attr["dateOfMonth"]}, #{attr["year"]}

샘플 결과

January 5, 2012

통화 예

구문

Price = #{attr["amount"]}#{attr["currency"]}

샘플 결과

2.50EUR

전자 메일 주소 예

구문

EmailAddress = #{attr["userName"]}@#{attr["domainName"]}

샘플 결과

JaneDoe@company.com

구문

AcmeEmailAddress = #{attr["AcmeIDKey"]}@acme.com

샘플 결과

bsmith@acme.com

신뢰 당사자 측에서 특성 매핑 구성

SiteMinder 가 어설션 특성에 적용할 수 있는 매핑 규칙 집합을 정의하십시오. SiteMinder 에서는 특정한 어설션 특성 또는 몇 가지 응용 프로그램 특성의 조합을 매핑할 수 있습니다. 매핑의 결과는 단일 응용 프로그램 특성이거나 여러 특성일 수 있습니다.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 "응용 프로그램 통합" 단계로 이동합니다.
2. "응용 프로그램 특성에 매핑" 섹션에서 "특성 매핑 사용" 확인란을 선택합니다.

   "응용 프로그램 특성 정의" 테이블이 표시됩니다.
3. 기존 응용 프로그램 특성을 수정하거나 테이블에서 새로 정의합니다. 모든 응용 프로그램 특성이 대상 응용 프로그램으로 전송됩니다.

   "어설션 특성" 열의 값 구문은 UEL(Unified Expression Language)을 준수해야 합니다.

   슬라이더 단추(<<)를 사용하여 식 작성기를 열고 사용 가능한 옵션을 표시합니다. 목록의 항목을 특성 값에 추가하려면 어설션 또는 특수 문자를 선택하고 "추가"를 클릭합니다.

   참고: "응용 프로그램 특성 테이블"에서 "쿠키 데이터" 및 특수 문자를 지정한 경우에는 "URL 인코드 특성 쿠키 데이터" 옵션을 선택하십시오. 확인란은 대화 상자의 "대상 응용 프로그램" 섹션에 있습니다. 특수 문자는 드롭다운 목록에서 추가하거나 직접 입력할 수 있습니다. 또한 대상 응용 프로그램은 수신되는 응용 프로그램 특성의 이름과 값을 URL 디코딩해야 합니다.
4. (선택 사항) 기본 매핑으로 부족한 경우 원하는 수의 행을 추가합니다.

   기본적으로 원격 생산자 또는 IdP 엔터티에서 정의된 모든 어설션 특성이 기본 매핑으로 테이블에 포함됩니다. 원래의 어설션 특성은 변경되지 않습니다. 이 매핑은 수정할 수 있습니다.
5. 응용 프로그램 특성이 대상 응용 프로그램으로 전송되는 방법을 구성합니다. "응용 프로그램 통합" 대화 상자의 "대상 응용 프로그램" 섹션에서 방법을 구성합니다.

특성 매핑 구성이 완료되었습니다.

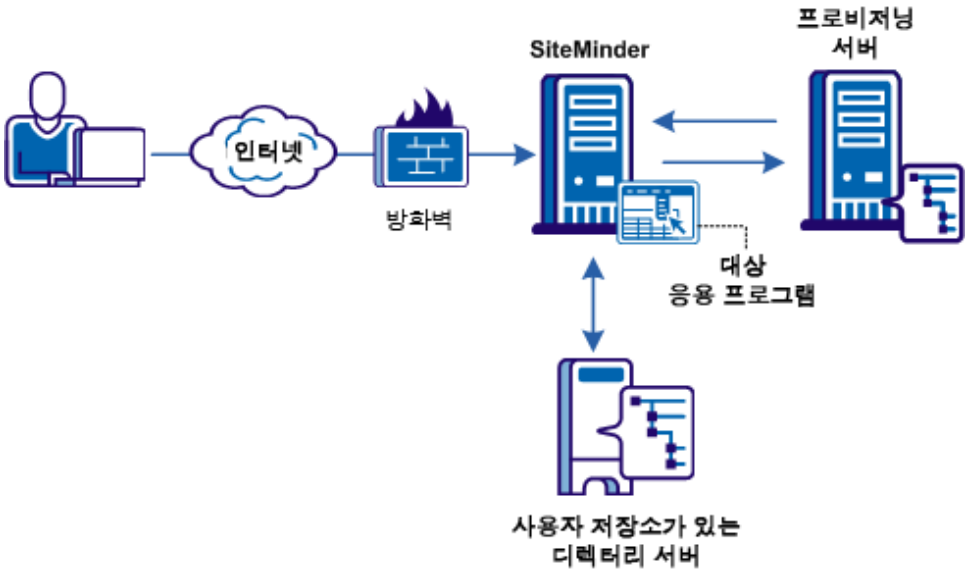
신뢰 당사자의 사용자 프로비저닝

페더레이션된 네트워크에서 신뢰 당사자는 다른 어설션 당사자에서 페더레이션되는 사용자에 대한 계정을 설정할 수 있습니다. 동적 프로비저닝은 데이터 및 응용 프로그램에 액세스하기 위해 필요한 계정 권한 및 액세스 권한을 가진 클라이언트 계정을 만드는 프로세스를 지원합니다.

원격 프로비저닝

원격 프로비저닝은 타사 프로비저닝 응용 프로그램을 사용하여 사용자 계정을 생성합니다. 그런 다음 응용 프로그램은 CA SiteMinder?Federation 가 있는 페더레이션 시스템의 정책 서버로 필요한 정보를 다시 전달합니다. 정책 서버는 데이터를 사용하여 사용자 자격 증명을 생성합니다.

원격 프로비저닝은 신뢰 당사자 측에서 수행됩니다. 다음 그림에서는 원격 프로비저닝 설정을 보여 줍니다.



상위 수준 프로비저닝 프로세스는 다음과 같습니다.

1. 신뢰 당사자의 정책 서버는 어설션과 함께 리소스에 대한 요청을 수신합니다. 하지만 사용자 디렉터리에서 사용자를 찾을 수 없습니다.
2. 프로비저닝이 사용되는 경우 정책 서버는 어설션 데이터가 포함된 활성 응답을 처리하고 어설션 데이터가 포함된 쿠키를 생성합니다. 또한 상태를 유지하는 쿠키가 생성되어 프로비저닝 요청이 수행되었음을 나타냅니다.
3. 브라우저는 개방 형식 쿠키 또는 헤더와 함께 프로비저닝 응용 프로그램으로 리디렉션됩니다.
4. 일반적으로 프로비저닝 응용 프로그램은 사용자에게 로그인하라는 메시지를 표시합니다. 사용자가 로그인하면 응용 프로그램은 쿠키나 헤더를 읽습니다. 응용 프로그램은 어설션 데이터와 로그인 자격 증명을 사용하여 사용자 계정을 설정합니다.

   프로비저닝 응용 프로그램은 CA SiteMinder® Federation Java 또는 .NET SDK 를 사용하여 개방 형식 쿠키를 소비할 수 있습니다.
5. 계정이 프로비저닝되면 브라우저는 사용자를 신뢰 당사자의 어설션 소비자 서비스로 다시 리디렉션합니다. 프로비저닝에 대한 상태 정보를 유지 관리하는 쿠키가 검사되어 사용자가 프로비저닝되었음이 확인됩니다. 자격 증명이 생성되어 인증 체계로 전달됩니다.

   참고: 프로비저닝 응용 프로그램은 신뢰 당사자에 있는 어설션 소비자 서비스의 URI 를 알아야 합니다. 예를 들어 신뢰 당사자 SiteMinder 의 SAML 2.0 URI 는 https://*sp_server:port*/affwebservices/public/saml2assertionconsumer 입니다.
6. 정책 서버는 사용자 명확성 확인 과정을 두 번째 시도합니다. 프로비저닝에 성공하면 사용자가 인증되고 쿠키 또는 헤더가 대상 응용 프로그램으로 전송됩니다.

   대상 응용 프로그램에 대해 선택한 리디렉션 모드에 따라 대상 응용 프로그램으로의 데이터 전달 방법이 결정됩니다.
7. 사용자가 대상 리소스로 리디렉션됩니다.

프로비저닝 응용 프로그램으로 어설션 데이터 전송

원격 프로비저닝을 수행하기 위해 SiteMinder 는 브라우저를 어설션 데이터와 함께 프로비저닝 응용 프로그램으로 리디렉션합니다.

SiteMinder 는 다음 방법 중 하나를 사용하여 어설션 데이터를 전달할 수 있습니다.

개방 형식 쿠키

SAML 어설션 정보를 개방 형식 쿠키로 전송합니다. 쿠키에는 어설션 데이터에 기반한 로그인 ID 가 포함됩니다.

참고: 개방 형식 쿠키를 사용하는 경우에는 SiteMinder 시스템과 원격 프로비저닝 시스템이 동일한 도메인에 있어야 합니다.

쿠키는 다음 두 가지 방법 중 하나로 만들 수 있습니다.

- CA SiteMinder® Federation SDK 에서 쿠키를 만듭니다.

  FIPS 알고리즘 중 하나(AES 알고리즘)를 선택하는 경우 CA SiteMinder® Federation SDK 를 사용하여 쿠키를 생성하십시오. .NET SDK 를 사용하려는 경우에는 AES128/CBC/PKCS5Padding 암호화 알고리즘만 사용하십시오. 프로비저닝 응용 프로그램이 .NET 을 사용하는 경우 프로비저닝 서버의 .NET SDK 가 개방 형식 쿠키를 읽습니다.

  프로비저닝 응용 프로그램은 쿠키를 만들기 위해 사용 중인 SDK 와 동일한 언어를 사용해야 합니다. CA SiteMinder® Federation Java SDK 를 사용하는 경우 응용 프로그램은 Java 로 작성되어야 합니다. .NET SDK 를 사용하는 경우 응용 프로그램이 .NET 을 지원해야 합니다.

- 개방 형식 쿠키를 수동으로 만듭니다.

  CA SiteMinder® Federation SDK 를 사용하지 않고 개방 형식 쿠키를 만들려면 원하는 프로그래밍 언어를 사용하십시오. 개방 형식 쿠키의 내용에 대한 자세한 내용을 참조하십시오.

  쿠키를 작성하는 언어는 UTF-8 인코딩을 지원하고 관리 UI 에서 선택할 수 있는 **PBE** 암호화 알고리즘 중 하나여야 합니다.

  쿠키를 암호화하기 위해 FIPS 호환(AES) 알고리즘을 선택하는 경우 프로비저닝 응용 프로그램은 SDK 를 사용하여 개방 형식 쿠키를 읽어야 합니다.

  브라우저에서 개방 형식 쿠키가 설정되었는지 확인하십시오.

개방 형식 쿠키 게시

개방 형식 쿠키 게시는 개방 형식 쿠키와 유사하지만 HTTP-POST 요청 형식으로 데이터를 보냅니다. 쿠키 데이터 제한으로 인해 데이터가 손실될 것을 우려하는 경우 이 옵션을 사용하십시오.

HTTP 헤더

SiteMinder 는 어설션 정보를 HTTP 헤더로 전달할 수도 있습니다. HTTP 헤더를 사용하는 경우 SiteMinder 시스템 및 원격 프로비저닝 시스템이 서로 다른 도메인에 있을 수 있습니다.

HTTP 헤더를 사용하여 어설션 데이터를 전달 (페이지 275)하는 방법과 헤더를 보호하는 방법에 대한 자세한 내용을 참조하십시오.

전송 옵션은 파트너 관계 마법사의 "응용 프로그램 통합" 단계에서 구성할 수 있습니다.

사용자가 프로비저닝 응용 프로그램으로 리디렉션된 후에는 SiteMinder 에서 더 이상 프로세스 제어권이 없습니다. 사용자 계정 프로비저닝이 시간이 많이 걸리는 프로세스인 경우 프로비저닝 응용 프로그램에서 이 상황을 처리해야 합니다. 예를 들어 이 응용 프로그램은 사용자에게 프로비저닝이 진행 중임을 알리는 메시지를 보낼 수 있습니다. 이 정보를 통해 사용자는 사용자 계정을 사용할 수 있을 때까지 로그인을 시도하지 않아야 한다는 점을 알 수 있습니다.

원격 프로비저닝 구성

원격 프로비저닝을 구성하려면 어설션 데이터에 대한 전송 옵션을 확인하고 프로비저닝 서버의 URL 을 제공하십시오.

원격 프로비저닝 구성 이외에도 "IDP 가 사용자 식별자를 만들도록 허용" 옵션을 선택할 수 있습니다. 이 옵션을 선택하면 사용자에 대한 식별자가 없는 경우 IdP 가 영구 식별자를 생성할 수 있습니다. 이 "허용/만들기" 기능은 로컬 방법에서는 필수지만 로컬 계정 연결을 사용한 프로비저닝에만 사용되는 것은 아닙니다.

IdP 가 다른 특성과 함께 전송되는 사용자 식별자를 생성하도록 하려면 원격 프로비저닝과 함께 "허용/만들기" 기능이 사용되도록 설정할 수 있습니다. 생성된 식별자를 사용하는 방법은 원격 프로비저닝 서버의 응용 프로그램에서 결정합니다. 응용 프로그램은 로컬 계정 연결을 수행할 수 있지만 SiteMinder 로컬 계정 연결은 수행할 수 없습니다.

원격 프로비저닝을 구성하려면

1. 파트너 관계 마법사의 "응용 프로그램 통합" 단계에서 시작합니다.
2. "사용자 프로비저닝" 섹션에서 프로비저닝 유형을 선택합니다.
3. 프로비저닝 유형으로 "원격"을 선택한 경우에는 표시되는 추가적인 필드에 데이터를 입력합니다.

   필드의 설명을 보려면 "도움말"을 클릭하십시오.
4. "확인" 단계를 선택하고 "마침"을 클릭하여 변경 내용을 저장합니다.

원격 프로비저닝 구성을 완료했습니다.

리디렉션 URL 을 사용하여 실패한 인증 처리(신뢰 당사자)

어설션 기반 인증은 어설션을 소비하는 사이트에서 실패할 수 있습니다. 인증이 실패하는 경우 추가 처리를 위해 사용자를 다른 응용 프로그램(URL)으로 리디렉션하도록 정책 서버를 구성할 수 있습니다. 예를 들어 사용자 명확성이 실패하는 경우 SiteMinder 가 사용자를 프로비저닝 시스템으로 보내도록 구성할 수 있습니다. 리디렉션 URL 설정은 선택 사항이며 신뢰 당사자 측에서만 구성할 수 있습니다.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 "응용 프로그램 통합" 단계에서 시작합니다.

   대화 상자의 "상태 리디렉션 URL" 섹션에서 원하는 특정 실패 조건에 대한 리디렉션만 지정합니다. SAML 2.0 의 경우 특정 HTTP 오류 조건에 대한 리디렉션도 구성할 수 있습니다.

   필드의 설명을 보려면 "도움말"을 클릭하십시오.
2. 구성하는 각 리디렉션 옵션에 대해 SiteMinder 가 사용자를 리디렉션하는 방법을 지정합니다. 옵션은 다음과 같습니다.

   302 데이터 없음(기본값)

   HTTP 302 리디렉션을 사용하여 데이터 없이 사용자를 리디렉션합니다.

   HTTP Post

   HTTP Post 프로토콜을 사용하여 사용자를 리디렉션합니다.

리디렉션 URL 의 구성이 완료되었습니다.

제 21 장: 파트너 관계 구성에 유용한 메타데이터 내보내기

이 섹션은 다음 항목을 포함하고 있습니다.

메타데이터 내보내기 개요

로컬 엔터티는 원격 엔터티가 손쉽게 엔터티를 생성하고 파트너 관계를 구성할 수 있도록 메타데이터를 생성합니다. 파트너 관계의 많은 요소가 메타데이터 파일에서 정의되므로 메타데이터를 사용하면 파트너 관계를 보다 효율적으로 구성할 수 있습니다. 원격 파트너는 메타데이터를 가져와서 메타데이터 문서의 정보를 기반으로 한 파트너 관계나 원격 엔터티를 생성할 수 있습니다.

기존의 로컬 어설션 엔터티 또는 신뢰 엔터티에서 메타데이터를 내보낼 수 있습니다.

관리 UI 에는 다음과 같이 메타데이터를 내보내기 위한 몇 가지 옵션이 있습니다.

- 로컬 엔터티에서 내보내기
- 로컬 파트너 관계에서 내보내기
- 로컬 WSFED 파트너 관계에 대한 메타데이터 교환

메타데이터를 보낼 때 파일을 사용하든 메타데이터 교환 프로필을 사용하든 관계없이 메타데이터를 가져오는 최종 목적은 동일합니다.

참고: SAML 1.1 의 경우 메타데이터 파일의 용어는 SAML 2.0 용어입니다. 이 규칙은 SAML 사양을 따릅니다. SAML 1.1 데이터를 가져올 때 용어는 SAML 1.1 용어를 사용하여 올바르게 가져오게 됩니다.

엔터티 수준 메타데이터 교환

로컬 엔터티에서 데이터를 내보낼 수 있습니다. 엔터티 수준에서 메타데이터를 내보낼 때는 내보내는 데이터에 파트너 관계 이름을 제공하십시오. 이 수준에서의 내보내기는 기본 파트너 관계 데이터를 정의합니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "페더레이션", "파트너 관계 페더레이션", "엔터티"를 선택합니다.
3. 목록에서 로컬 항목 옆의 "작업" 풀다운 메뉴를 클릭하고 "메타데이터 내보내기"를 선택합니다.

   "메타데이터 내보내기" 대화 상자가 열립니다.
4. 새 파트너 관계 이름을 지정합니다. 내보내기를 통해 생성된 메타데이터 파일에는 기본 파트너 관계를 설정하기 위한 정보가 포함되어 있습니다.
5. 대화 상자의 나머지 필드를 채웁니다. 대화 상자의 "메타데이터 내보내기 옵션" 섹션에서 설정을 입력하십시오.

   참고: 필드의 설명을 보려면 "도움말"을 클릭하십시오.
6. "Export"(내보내기)를 클릭합니다.
7. 메타데이터 파일을 열지 아니면 저장할지 묻는 대화 상자가 표시됩니다.

   파일을 열기만 하여 표시합니다.
8. 데이터를 로컬 시스템의 XML 파일에 저장합니다.

메타데이터가 지정된 XML 파일로 내보내졌습니다. 이 파일을 모든 파트너로 전송할 수 있습니다.

파트너 관계 수준 메타데이터 교환

로컬 파트너 관계에서 데이터를 내보낼 수 있습니다. 이 수준에서의 내보내기는 기본 파트너 관계 데이터를 정의합니다.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"를 선택합니다.

3. 목록의 파트너 관계 옆에 있는 "작업" 풀다운 메뉴를 선택합니다.
4. "메타데이터 내보내기"를 선택합니다.

   "메타데이터 내보내기" 대화 상자가 열립니다.
5. 정보를 검토합니다. 내보내기를 통해 생성된 메타데이터 파일에는 기본 파트너 관계를 설정하기 위한 정보가 포함되어 있습니다.
6. "메타데이터 내보내기 옵션" 섹션에서 메타데이터 문서의 서명 및 유효성 검사를 위한 설정을 완료합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
7. "Export"(내보내기)를 클릭합니다.
8. 메타데이터 파일을 열지 아니면 저장할지 묻는 대화 상자가 표시됩니다.

   파일을 열기만 하여 표시합니다.
9. 데이터를 로컬 시스템의 XML 파일에 저장합니다.

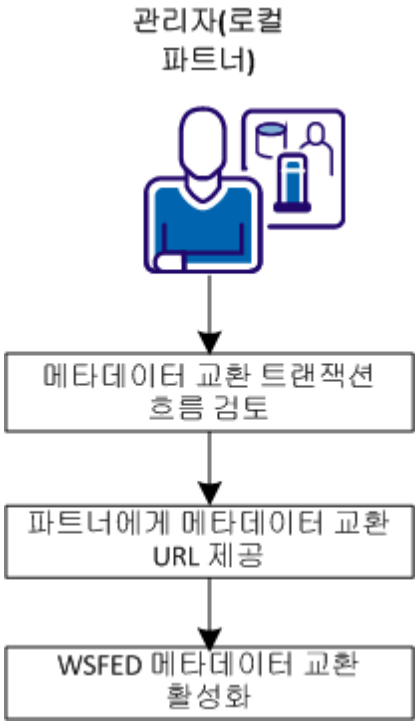
메타데이터가 지정된 XML 파일로 내보내졌습니다. 이 파일을 모든 파트너로 전송할 수 있습니다.

WS-페더레이션 메타데이터 교환이 사용되도록 설정하는 방법

정책 서버는 WS-페더레이션 파트너 관계에 대해 웹 서비스 메타데이터 교환 프로필을 지원합니다. 이 웹 서비스를 사용하면 SiteMinder 로컬 파트너가 원격 파트너의 메타데이터 요청에 응답할 수 있습니다. 교환은 HTTP 요청 및 응답의 형태로 이루어집니다.

HTTP 프로토콜을 사용하면 원격 엔터티가 프로그래밍 방식으로 페더레이션을 구성할 수 있습니다. 응용 프로그램에서 URL 을 사용하여 필요한 정보를 수집할 수 있습니다.

다음 그림에서는 메타데이터 교환을 위한 구성 단계를 보여 줍니다.



메타데이터 교환을 위한 다음 구성을 완료하십시오.

1. 메타데이터 교환 트랜잭션 흐름을 검토합니다. (페이지 293)
2. 메타데이터 교환 URL 을 파트너에게 제공합니다. (페이지 293)
3. WSFED 메타데이터 교환이 사용되도록 설정합니다 (페이지 294).

메타데이터 교환 트랜잭션 흐름

메타데이터 교환 트랜잭션의 프로세스 흐름은 다음과 같습니다.

1. 로컬 파트너가 제공한 메타데이터 교환 URL 로 원격 파트너가 요청을 보냅니다.
2. 로컬 파트너가 HTTP 응답에서 원격 파트너로 메타데이터를 다시 보냅니다. 정책 서버가 응답에 서명하여 메타데이터를 보호합니다. 원격 파트너가 응답을 확인하는 데 사용할 수 있는 인증서는 응답에 포함되어 있습니다.

   정책 서버는 요청이 있을 때 메타데이터 문서를 생성합니다. 이 문서는 로컬 파트너에 저장되지 않습니다.
3. 원격 파트너가 응답의 서명을 확인합니다. 서명이 유효하면 원격 파트너는 메타데이터 문서를 구문 분석하고 해당 정보를 사용하여 엔터티와 파트너 관계를 설정합니다.

파트너에 메타데이터 교환 URL 제공

메타데이터 트랜잭션이 발생하기 전에 원격 파트너에 메타데이터 교환 요청을 위한 URL 을 제공하십시오. 페더레이션된 파트너는 다음 URL 로 요청을 보내야 합니다.

https://*server:port*/affwebservices/public/FederationMetadata/*partnership_name*

server:port

메타데이터 교환 서비스를 호스트하는 시스템의 이름입니다.

partnership_name

구성된 파트너 관계의 이름입니다.

WSFED 메타데이터 교환이 사용되도록 설정

로컬 WS-페더레이션 파트너에서 메타데이터 교환 기능이 사용되도록 설정하십시오.

다음 단계를 수행하십시오.

1. 관리 UI 에 로그인합니다.
2. 수정할 WSFED 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "파트너 관계 구성" 단계에서 "메타데이터 교환 사용" 확인란을 선택합니다.
4. "확인" 단계로 이동하고 "마침"을 클릭합니다.
5. 기본 "파트너 관계 페더레이션" 탭으로 돌아갑니다("페더레이션", "파트너 관계 페더레이션).
6. 왼쪽 창에서 "메타데이터 교환 구성"을 선택합니다.

   "메타데이터 교환 구성" 화면이 표시됩니다.
7. 응답에 서명할 값을 제공합니다.
8. "저장"을 클릭합니다.

파트너 관계에 대한 메타데이터 교환이 구성되었습니다.

제 22 장: 문제 해결에 유용한 로그 파일

이 섹션은 다음 항목을 포함하고 있습니다.

페더레이션 추적 로깅

페더레이션 웹 서비스(FWS) 추적 로깅 기능과 정책 서버 프로파일러는 페더레이션 서비스의 성능을 모니터링합니다. 이러한 로깅 메커니즘은 시스템 성능을 분석하고 문제를 해결할 수 있도록 페더레이션된 작업에 대한 정보를 제공합니다.

페더레이션 프로세스에 대한 세부 정보를 추출하려면 웹 에이전트 옵션 팩과 정책 서버가 설치된 곳에서 추적 로깅을 활성화하십시오. 예를 들어 FWSTrace.log 를 살펴보면 생성된 SAML 어설션을 확인하거나 현재 사용자의 이름을 수집할 수 있습니다.

참고: 성능에 영향을 줄 수 있으므로 추적 메시지는 일반 작업 중 일반적으로 꺼져 있습니다.

수집된 추적 메시지는 두 개의 추적 로그에 기록됩니다.

FWSTrace.log

FWSTrace.log 는 웹 에이전트 옵션 팩이 설치 또는 배포된 웹 서버 또는 응용 프로그램 서버의 /log 디렉터리에 있습니다.

웹 서버

webagent/log

webagent_optionpack/log

응용 프로그램 서버

default_deployment_directory/log

SPS 페더레이션 게이트웨이

sps_home/secure-proxy/proxy-engine/logs

smtracedefault.log

smtracedefault.log 는 *smtracedefault.log*/log 디렉터리에 있습니다.

siteminder_home 은 제품의 설치 디렉터리를 나타냅니다.

FWSTrace.log 및 smtracedefault.log 에는 트랜잭션 중 발생하는 사항을 나타내는 검사점 로그 메시지가 있습니다. 예를 들면 다음과 같습니다.

```
[07/30/2013][11:34:44][4260][5824][1181adbb-993f775c-33ba08f3-76b5
2f3b-3d2280cd-4ae][SSO.java][processRequest][Reading SAML 2.0 SP
Configuration [CHECKPOINT = SSOSAML2_SPCONFREAD_REQ]
```

트랜잭션 중 발생하는 일부 프로세스를 추적하기 위해 이러한 검사점 메시지를 검색할 수 있습니다.

검사점 메시지에 추가하여, 트랜잭션을 추적하기 위해 로그에서 트랜잭션 ID 를 추적할 수 있습니다. 트랜잭션이 실패하면 검사점 메시지 및 트랜잭션 ID 는 특정 문제를 파악하는 데 도움을 줄 수 있습니다.

페더레이션 문제 해결에 도움이 되는 트랜잭션 ID

한 파일 내에 수많은 트랜잭션이 기록되어 있는 경우 페더레이션 트랜잭션의 문제를 해결하는 것이 쉽지 않습니다. 트랜잭션 로그에서 단일 트랜잭션을 추적하려면 SAML 트랜잭션 ID 를 사용하십시오. 페더레이션 호출이 발생하면 FWS 응용 프로그램이 먼저 SAML 트랜잭션 ID 를 생성합니다. SAML 트랜잭션 ID 는 한 번만 생성됩니다. 이 고유 SAML 트랜잭션 ID 는 여러 트랜잭션 ID 로 매핑될 수 있습니다.

예를 들어 SAML 2.0 POST 트랜잭션용 fwstrace.log 에서 다음 메시지가 나타날 수 있습니다. 두 트랜잭션 ID 의 매핑을 보여 주는 굵게 표시된 행을 참고하십시오.

```
[08/01/2013][17:33:54][2292][1884][1c2d7650-b006e46a-ed071f41-bbbe
de33-fe78e2dd-38d][SSO.java][processAuthentication][SAMLTransactio
nID 2aaf90ec-fdef4897-0ef49d91-63d4031d-f508a3e9-12 maps to
TransactionID: 1c2d7650-b006e46a-ed071f41-bbbede33-fe78e2dd-38d.]
```

CA SiteMinder?Federation 시스템은 어설션 당사자로 기능하는 경우에만 새 SAMLTransactionID 를 생성합니다. 이러한 특정 활동은 다음의 경우 발생합니다.

- 페더레이션 웹 서비스가 세션을 구성하기 위해 브라우저를 인증 URL 로 리디렉션하는 경우
- 다음 HTTP-아티팩트 싱글 사인온 트랜잭션의 경우:
  - 어설션 당사자가 신뢰 당사자에 아티팩트를 반환하는 경우
  - 어설션 당사자가 아티팩트를 확인하는 경우
- 사용자가 아이덴티티 검색 프로필 URL 로 리디렉션되는 경우
- 어설션 당사자에서 싱글 로그아웃 중

신뢰 당사자에는 요청 ID 가 있으며, 이 ID 는 로그 파일을 통해 쉽게 추적될 수 있습니다. 요청 ID 는 신뢰 당사자에서 SAMLTransactionID 를 생성하기 위해 CA SiteMinder?Federation 시스템에서 필요하지 않습니다.

각 고유 SAML 트랜잭션 ID 에는 여러 트랜잭션 ID 가 있을 수 있습니다. 새 HTTP 트랜잭션이 발생하면 새 트랜잭션 ID 가 생성됩니다. 이 트랜잭션 ID 는 단일 SAML 트랜잭션 ID 로 매핑됩니다. 예를 들어, 추적 로그에서 다음과 같은 항목을 볼 수 있습니다.

```
SamlTransactionID ["xyz"] maps to TransationID["123"]
["123"] HTTP operation
["123"] HTTP operation
```

A new transaction ID "456" is generated:

```
SamlTransactionID["xyz"] Maps to Transactionid["456"]
["456"] <some operation>
["456"] <some operation>
```

트랜잭션 ID 는 fwstrace.log 및 smtracedefault.log 에 배치됩니다. 즉, 단일 트랜잭션에 대해 동일한 트랜잭션 ID 집합이 이들 로그에 각각 기록됩니다. 이들 로그에서 ID 를 추적하면 트랜잭션을 추적할 수 있습니다. 오류가 발생한 경우 ID 를 사용하여 오류가 발생한 해당 트랜잭션에서 어느 이벤트가 실패했는지 확인할 수 있습니다.

로그에서 단일 트랜잭션을 추적하는 방법

트랜잭션을 모니터링하려면 FWSTrace.log 또는 smtracedefault.log 에서 트랜잭션 ID 의 두 가지 유형을 추적할 수 있습니다. 오류가 발생하는 경우 ID 를 확인하면 오류 지점을 파악하는 데 도움이 될 수 있습니다.

로그에서 트랜잭션을 추적하려면 다음 방법 중 하나 이상을 사용하십시오.

- 텍스트 편집기에서 추적 파일을 열고 문자열 **SAMLTransactionID**(공백 없음) 또는 특정 SAMLTransactionID 를 검색하십시오. 로그에 있는 이 항목의 모음을 통해 전체 엔드-투-엔드 트랜잭션을 볼 수 있습니다. 트랜잭션이 진행된 정도를 파악할 수 있습니다.
- 로그 파일에서 트랜잭션 ID 를 추적하십시오. 트랜잭션 ID 는 HTTP 트랜잭션을 나타냅니다. 하나의 SAML 트랜잭션 ID 에 여러 트랜잭션 ID 가 연결될 수 있습니다. 실패한 트랜잭션은 브라우저에 트랜잭션 ID 를 표시합니다. FWSTrace.log 및 smtracedefault 로그에서 검사점 오류 메시지를 검색하려면 표시된 트랜잭션 ID 를 사용하십시오.
- 파일을 검색하는 도구를 사용하여 로그 파일을 구문 분석하십시오. UNIX 및 Windows 플랫폼에서는 grep 명령과 같은 도구를 사용할 수 있습니다. grep 명령은 원시 데이터 스트림을 한 줄씩 표시하므로 크기가 큰 텍스트 파일을 텍스트 편집기에 로드할 필요가 없습니다.

  예:

```
[usr@rhel632 etc]# more fwstrace.log| grep checkpoint
[CHECKPOINT = SSOSAML2_SPCONFFROMPS_REQ]]
[CHECKPOINT = SSOSAML2_SPCONFREAD_REQ]]
[CHECKPOINT = SSOSAML2_SPCONFFROMCACHE_REQ]]
[CHECKPOINT = SSOSAML2_SESSIONCOOKIEVALIDATE_REQ]]
```

페더레이션 서비스 추적 로깅(smtracedefault.log)

프로파일러는 로깅용 정책 서버 기능입니다. 프로파일러를 사용하여 페더레이션 서비스에 대한 추적 메시지를 수집하여 smtracedefault.log 파일에 기록할 수 있습니다.

정책 서버에서 페더레이션 서비스에 대한 추적 메시지를 제어하는 구성 요소는 Fed_Server 구성 요소입니다.

정책 서버 프로파일러를 사용하여 내부 정책 서버 진단 및 처리 기능을 추적할 수 있습니다.

다음 단계를 수행하십시오.

1. 정책 서버 관리 콘솔을 시작합니다.

   중요! Windows Server 2008 에서 이 그래픽 사용자 인터페이스에 액세스하는 경우에는 관리자 권한을 사용하여 바로 가기를 여십시오. 관리자로 시스템에 로그인한 경우에도 관리자 권한을 사용하십시오. 자세한 내용은 SiteMinder 구성 요소의 릴리스 정보를 참조하십시오.

2. "프로파일러" 탭을 클릭합니다.
3. 프로파일링을 사용하도록 "프로파일링 사용" 옵션을 설정합니다.
4. 프로파일러의 구성 설정을 선택하려면 다음 중 하나를 수행합니다.
   - "구성 파일" 드롭다운 목록에 있는 기본 smtracedefault.txt 파일에 지정된 프로파일러 설정을 그대로 사용합니다.
   - "구성 파일" 드롭다운 목록에서 이 관리 세션 중에 이미 선택된 다른 구성 파일을 선택합니다.
   - "찾아보기" 단추를 클릭하여 다른 구성 파일을 선택합니다.
5. 프로파일러 구성 파일에 저장된 프로파일러 설정을 변경한 후 동일한 파일이나 새 파일에 저장하려면 "설정 구성" 단추를 클릭하여 "정책 서버 프로파일러" 대화 상자를 엽니다.
6. "출력" 그룹 상자에 있는 설정을 조정하여 정책 서버 프로파일러에서 생성되는 정보의 출력 형식을 지정합니다.
7. "적용"을 클릭하여 변경 내용을 저장합니다.

참고:

프로파일러 설정에 대한 변경 내용은 자동으로 적용됩니다. 그러나 정책 서버를 다시 시작하면 새 출력 파일이 생성됩니다(프로파일러의 출력 형식이 파일로 구성된 경우). 기존 프로파일러 출력 파일은 버전 번호를 포함하여 자동으로 저장됩니다. 예를 들면 다음과 같습니다.

```
smtracedefault.log.1
```

Windows 에서 콘솔 로깅을 사용하거나 사용하지 않도록 설정할 때처럼 로깅 또는 추적 기능 설정에 대해 변경한 내용이 프로파일 출력 파일과 관련이 없는 경우 하나의 파일 버전이 저장되지 않고 기존 파일에 새 출력이 추가됩니다.

기본적으로 정책 서버는 최대 10 개의 출력 파일(현재 파일과 백업 파일 9 개)을 보존합니다. 파일 제한 10 개에 도달하면 오래된 파일부터 자동으로 최신 파일로 대체됩니다. TraceFilesToKeep DWORD 레지스트리 설정을 필요한 10 진수 값으로 구성하여 보존할 파일 수를 변경할 수 있습니다. TraceFilesToKeep 레지스트리 설정은 다음 레지스트리 위치에 생성해야 합니다.

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Netegrity\SiteMinder\CurrentVersion\
LogConfig\TraceFilesToKeep
```

"프로파일러" 탭에 있는 "추적 버퍼링" 옵션은 정책 서버 성능을 높이기 위해 기본적으로 설정되어 있습니다. 이 옵션은 Solaris 시스템에만 해당됩니다.

페더레이션 웹 서비스 추적 로깅(FWSTrace.log)

추적 데이터 수집 태스크를 더 단순화하기 위해 일련의 미리 구성된 템플릿이 웹 에이전트 옵션 팩과 함께 설치됩니다. 자체적으로 추적 구성 파일을 만들지 않고 이러한 템플릿을 사용하여 데이터를 수집할 수 있습니다.

다음과 같은 템플릿을 사용할 수 있습니다.

템플릿	수집되는 추적 메시지
FWSTrace.conf	기본 템플릿. 지정하는 데이터를 수집합니다.
FWS_SSOTrace.conf	싱글 사인온 메시지를 수집합니다.
FWS_SLOTrace.conf	싱글 로그아웃 메시지를 수집합니다.
FWS_IPDTrace.conf	아이덴티티 공급자 검색 프로필 메시지를 수집합니다.

이 모든 템플릿에는 추적 중인 특정 데이터에 대한 Fed_Client 구성 요소와 하위 구성 요소가 포함되어 있습니다. 정확한 콘텐츠를 보려면 각 템플릿을 여십시오.

다음 단계를 수행하십시오.

1. *web_agent* 또는 *web_agent_option_pack_home*/config 에 있는 템플릿 디렉터리로 이동합니다.
2. 템플릿의 복사본을 만든 다음 이름을 변경합니다.
3. (선택 사항) 모니터링할 데이터만 포함하도록 템플릿을 수정합니다.

   참고: 템플릿을 직접 편집하지 마십시오.
4. 새 템플릿을 저장합니다.

이 템플릿은 페더레이션 시스템이 모니터링하는 페더레이션 구성 요소를 결정합니다. 추적 로깅을 활성화하고 로그 파일에 데이터가 표시되는 형식을 지정하려면 Logger.Config 속성 파일을 수정하십시오.

다음 단계를 수행하십시오.

1. *web_agent* 또는 *webagent_optionpack_home*/affwebservices/WEB-INF/classes 로 이동합니다.
2. LoggerConfig.properties 파일을 엽니다. LoggerConfig.properties 파일에는 이 모든 설정에 대한 설명이 포함되어 있습니다.
3. TracingOn 설정을 Yes 로 설정합니다. 이 옵션은 로그 파일에 메시지를 쓰도록 추적 기능에 지시합니다.
4. TraceFileName 설정을 로그 파일의 전체 경로로 설정합니다. 기본 위치는 *web_agent or webagent_optionpack_home*/config/FWSTrace.log 입니다.

   참고: 이 로그 파일의 이름은 변경할 수 있습니다. 기본 이름은 FWSTrace.log 입니다.
5. TraceConfigFile 설정을 추적 구성 파일의 전체 경로로 설정합니다. 이 파일은 기본 템플릿, 다른 미리 구성된 템플릿 중 하나, 자체적인 구성 파일일 수 있습니다. 지정하는 템플릿에 관계없이 모든 출력은 TraceFileName 설정에 지정하는 로그 파일에 기록됩니다.

   단 하나의 템플릿만 지정하십시오. 모든 템플릿은 *web_agent* 또는 *web_agent_option_pack_home*/config 디렉터리에 저장됩니다.

6. 또는, 추적 로그 출력의 정보가 표시되는 방식을 수정합니다. 다음 설정은 로그 파일의 형식을 결정합니다.
   - TraceRollover
   - TraceSize
   - TraceCount
   - TraceFormat
   - TraceDelim

FWS 템플릿 샘플

다음 텍스트는 FWS_SLOTrace.conf 템플릿에서 발췌한 내용입니다. 파일의 대부분에는 파일 사용 방법, 명령 구문 및 Fed_Client 구성 요소에 대해 사용 가능한 하위 구성 요소에 대한 설명과 지침이 포함되어 있습니다.

발췌한 내용에서는 모니터링되는 구성 요소 Fed_Client 와 하위 구성 요소(Single_Logout 및 Configuration)를 보여 줍니다. 각 메시지의 필수 콘텐츠를 나타내는 특정 데이터 필드(Date, Time, Pid, Tid, TransactionId, SrcFile, Function, Message)도 보여 줍니다.

```
components: Fed_Client/Single_Logout, Fed_Client/Configuration
data: Date, Time, Pid, Tid, TransactionID, SrcFile, Function, Message
```

제 23 장: 개방 형식 쿠키 정보

페더레이션 개방 형식 쿠키를 사용하면 응용 프로그램은 SiteMinder 에 사용자 특성을 어설션하고 SiteMinder 가 캡슐화하는 사용자 특성을 사용할 수 있습니다. 개방 형식 쿠키의 일반적인 특성은 다음과 같습니다.

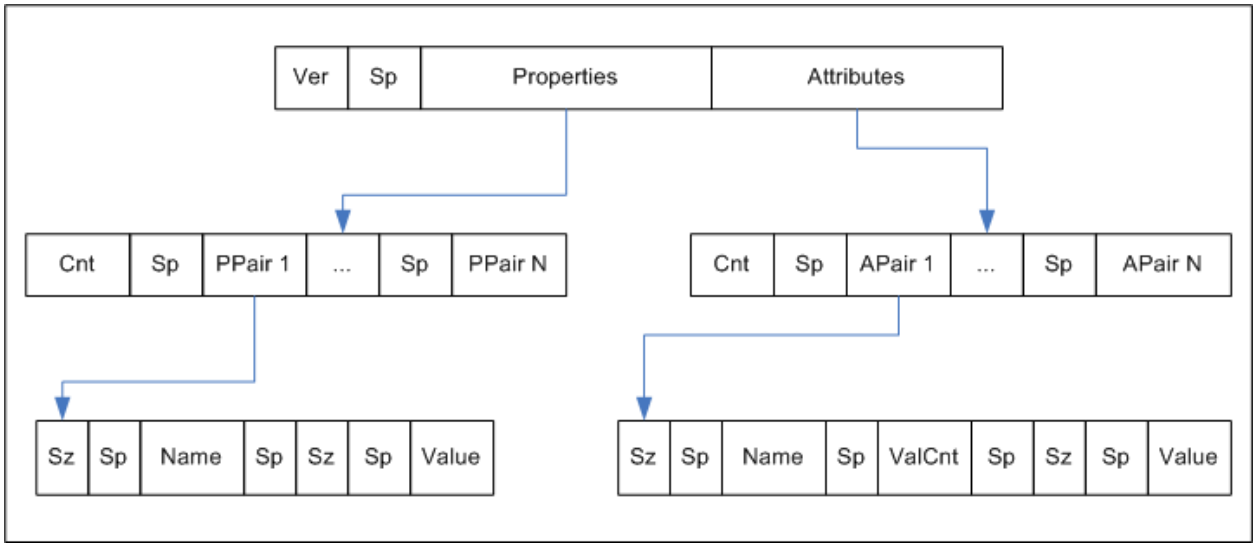
- 모든 프로그래밍 언어로 작성된 응용 프로그램이 쿠키를 사용할 수 있습니다.
- 쿠키 내용은 국제 문자 집합을 지원하는 UTF-8 바이트 문자열로 구성됩니다.
- 이름/값 쌍의 앞에 각 이름/값 쌍의 UTF-8 바이트 결합 크기가 나옵니다.
- 읽기 쉽도록 공백 문자가 추가됩니다.
- 쿠키는 구문 분석이 간단하며 쉽게 확장할 수 있습니다.

중요! 쿠키에 '='와 같은 안전하지 않은 문자가 포함될 경우 값을 큰따옴표로 묶으십시오. 사용자 인터페이스 또는 SDK 를 통해 이 옵션을 지정할 수 있습니다.

개방 형식 쿠키에는 다음의 속성 정보가 포함됩니다.

- 쿠키 버전
- 이름 ID
- 이름 ID 형식
- 세션 ID
- AuthnContext
- UserDN(사용자 ID 와 동일)

다음 다이어그램에서는 개방 형식을 보여 줍니다.



키:

- Ver - 쿠키 형식 버전. CA SiteMinder® Federation r12.1 의 경우 이 값은 1 입니다.
- Sp - ASCII 공백 문자. 가독성 향상의 목적으로만 사용됩니다.
- Properties - 프린서펄에 대한 정보입니다.
- Attributes - 어설션의 SAML 특성
- Cnt - 뒤에 나오는 이름 값 쌍의 수이며 ASCII 로 표현됩니다.
- Sz - 뒤에 나오는 이름 또는 값의 길이
- ValCnt - 뒤에 나오는 특성 값의 수입니다. CA SiteMinder® Federation r12.1 의 경우 특성에 대한 여러 개의 값이 지원됩니다. 이 값을 1 로 설정합니다.

이 형식의 BNF(Backus-Naur Form)는 다음과 같습니다(0*는 0 이상, 1*은 최소 1 을 의미).

- DIGIT = ASCII 숫자(0~9)
- CHAR = UTF-8 문자
- Sp = ASCII 공백(문자 32)

- Token = 1*CHAR
- Cookie = 버전 Sp 속성 특성
- Version = 1*DIGIT
- Cnt = 1*DIGIT
- Properties = Cnt 1*PPair
- Attributes = Cnt 0*APair
- ValCnt = 1*DIGIT
- PPair = Sz Sp 이름 Sp Sz Sp 값
- APair = Sz Sp 이름 Sp Sz Sp 값
- Sz = 1*DIGIT
- Name = 토큰

Value = 토큰

개방 형식 쿠키의 내용

페더레이션 개방 형식 쿠키를 사용하면 응용 프로그램은 SiteMinder 에 사용자 특성을 어설션하고 SiteMinder 가 캡슐화하는 사용자 특성을 사용할 수 있습니다. 개방 형식 쿠키의 일반적인 특성은 다음과 같습니다.

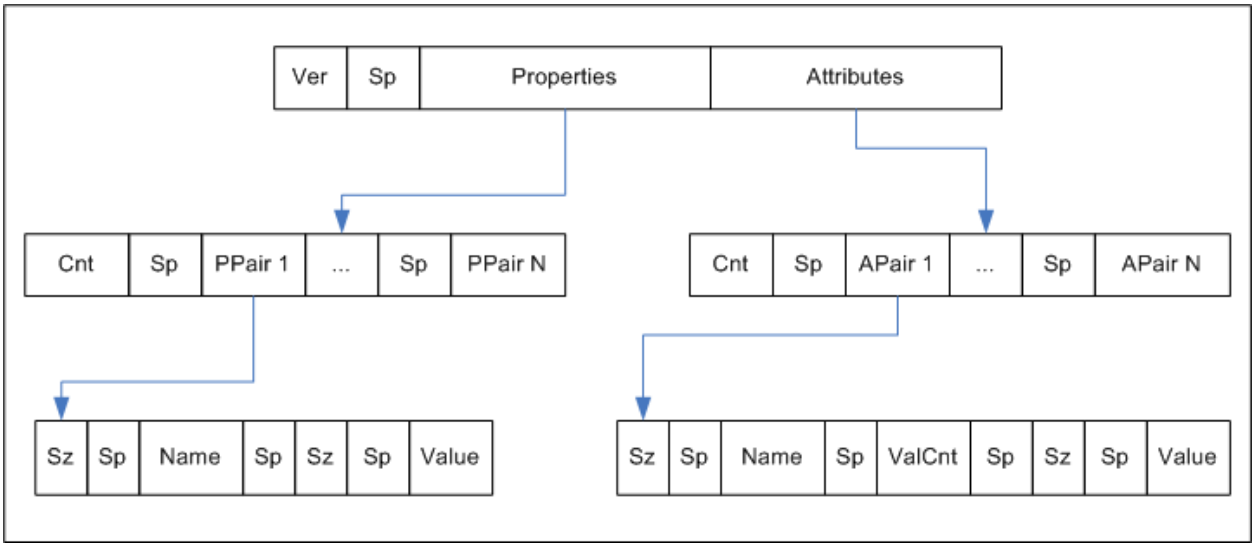
- 모든 프로그래밍 언어로 작성된 응용 프로그램이 쿠키를 사용할 수 있습니다.
- 쿠키 내용은 국제 문자 집합을 지원하는 UTF-8 바이트 문자열로 구성됩니다.
- 이름/값 쌍의 앞에 각 이름/값 쌍의 UTF-8 바이트 결합 크기가 나옵니다.
- 읽기 쉽도록 공백 문자가 추가됩니다.
- 쿠키는 구문 분석이 간단하며 쉽게 확장할 수 있습니다.

중요! 쿠키에 '='와 같은 안전하지 않은 문자가 포함될 경우 값을 큰따옴표로 묶으십시오. 사용자 인터페이스 또는 SDK 를 통해 이 옵션을 지정할 수 있습니다.

개방 형식 쿠키에는 다음의 속성 정보가 포함됩니다.

- 쿠키 버전
- 이름 ID
- 이름 ID 형식
- 세션 ID
- AuthnContext
- UserDN(사용자 ID 와 동일)

다음 다이어그램에서는 개방 형식을 보여 줍니다.



키:

- Ver - 쿠키 형식 버전. CA SiteMinder® Federation r12.1 의 경우 이 값은 1 입니다.
- Sp - ASCII 공백 문자. 가독성 향상의 목적으로만 사용됩니다.
- Properties - 프린서펄에 대한 정보입니다.
- Attributes - 어설션의 SAML 특성
- Cnt - 뒤에 나오는 이름 값 쌍의 수이며 ASCII 로 표현됩니다.
- Sz - 뒤에 나오는 이름 또는 값의 길이
- ValCnt - 뒤에 나오는 특성 값의 수입니다. CA SiteMinder® Federation r12.1 의 경우 특성에 대한 여러 개의 값이 지원됩니다. 이 값을 1 로 설정합니다.

이 형식의 BNF(Backus-Naur Form)는 다음과 같습니다(0*는 0 이상, 1*은 최소 1 을 의미).

- DIGIT = ASCII 숫자(0~9)
- CHAR = UTF-8 문자
- Sp = ASCII 공백(문자 32)
- Token = 1*CHAR
- Cookie = 버전 Sp 속성 특성
- Version = 1*DIGIT
- Cnt = 1*DIGIT
- Properties = Cnt 1*PPair
- Attributes = Cnt 0*APair
- ValCnt = 1*DIGIT
- PPair = Sz Sp 이름 Sp Sz Sp 값
- APair = Sz Sp 이름 Sp Sz Sp 값
- Sz = 1*DIGIT
- Name = 토큰
- Value = 토큰

부록 A: 암호화 및 암호 해독 알고리즘

이 섹션은 다음 항목을 포함하고 있습니다.

개방 형식 쿠키 암호화 알고리즘

개방 형식 쿠키는 암호 기반 암호화에 대해 다음 옵션을 지원합니다.

FIPS_Compat 및 FIPS_Migration 모드

PBE/SHA1/AES/CBC/PKCS12PBE-1000-128

PBE/SHA1/AES/CBC/PKCS12PBE-1000-192

PBE/SHA1/AES/CBC/PKCS12PBE-1000-256

PBE/SHA256/AES/CBC/PKCS12PBE-1000-128

PBE/SHA256/AES/CBC/PKCS12PBE-1000-192

PBE/SHA256/AES/CBC/PKCS12PBE-1000-256

PBE/SHA1/3DES_EDE/CBC/PKCS12PBE-1000-3

PBE/SHA256/3DES_EDE/CBC/PKCS12PBE-1000-3

FIPS_Only 모드

AES128/CBC/PKCS5Padding

AES192/CBC/PKCS5Padding

AES256/CBC/PKCS5Padding

3DES_EDE/CBC/PKCS5Padding

디지털 서명 및 개인 키 알고리즘

SiteMinder 는 파트너 관계 서명 옵션에 다음 알고리즘을 사용합니다.

암호화 키 알고리즘

RSA-V15, RSA-OEAP

암호화 블록 알고리즘

3DES, AES-128, AES-256

SiteMinder 는 개인 키 생성에(인증서/키) 다음 알고리즘을 사용합니다.

키 알고리즘

RSA

서명 알고리즘

MD5withRSA, SHA1withRSA, SHA256withRSA 및 SHA512withRSA

백 채널 통신 알고리즘

HTTP-아티팩트 싱글 사인온 및 SAML 2.0 싱글 로그아웃에 관련된 백 채널 통신에 대해 SiteMinder 는 FIPS 모드에 따라 다음의 암호화를 지원합니다.

FIPS_Compat 및 FIPS_Migration 모드 - RC4 및 AES

RSA_With_RC4_SHA

RSA_With_RC4_MD5

RSA_With_AES_128_CBC_SHA

RSA_With_AES_256_CBC_SHA

FIPS_Only Mode - AES 만

RSA_With_AES_128_CBC_SHA

RSA_With_AES_256_CBC_SHA

Java SDK 암호화 알고리즘

CA SiteMinder?Federation Java SDK 는 다음의 암호화 알고리즘을 지원합니다.

암호 없음

"AES/CBC/PKCS5Padding"

암호 사용

"PBE/SHA1/AES/CBC/PKCS12PBE-5-128"

Crypto 알고리즘

FMCrypto 암호화/암호 해독 알고리즘

AES_128